



PlayBloodBowl

https://github.com/DavidPerezVelilla/TFG_DavidPerezVelilla

Proyecto final de David Pérez Velilla

2º Desarrollo de Aplicaciones Web

Tutor: Luis Miguel Morillas

CPIFP Los Enlaces, Zaragoza

Curso 2021/2022

Índice

1. Descripción general del proyecto.....	pág. 2
2. Acuerdo del proyecto.....	pág. 3
a. Requisitos funcionales y no funcionales.....	pág. 3
b. Tareas.....	pág. 8
c. Metodología.....	pág. 9
d. Planificación temporal de las tareas.....	pág. 10
e. Presupuesto.....	pág. 11
3. Análisis y diseño.....	pág. 12
a. Diagrama entidad - relación.....	pag. 12
b. Diagrama de clases.....	pág. 13
c. Casos de uso.....	pag. 14
d. Mockups.....	pág. 20
e. Herramientas y tecnologías.....	pag. 24
f. Arquitectura de componentes.....	pág. 27
4. Documento de implementación, pruebas e implantación.....	pág. 29
a. Implementación.....	pág. 29
b. Pruebas.....	pág. 30
c. Instalación y configuración.....	pág. 33
d. Manual del usuario.....	pág. 34
5. Documento de cierre.....	pág. 47
a. Resultados obtenidos y conclusiones.....	pág. 47
b. Diario de bitácora.....	pág. 48
6. Bibliografía.....	pag. 49

1. Descripción del proyecto

Contexto del proyecto

El proyecto “Play Blood Bowl” es una app web que permite a los usuarios la posibilidad de crear plantillas de equipo para poder jugar a este juego de mesa llamado Blood Bowl, una especie de “fútbol americano” basado en un universo de fantasía (universo de Warhammer).

En ella podrán consultar los jugadores de cada raza, estadísticas, habilidades, atributos y crear su equipo a medida. Además podrán consultar el reglamento, últimas novedades, tiendas de Zaragoza donde poder comprar las miniaturas y todo lo necesario para iniciarse.

Play Blood Bowl está enfocado al jugador de juego de mesa, por lo general hombre a partir de los 30 años de edad, con un nivel adquisitivo medio/alto. Un mercado en alza que con la pandemia ha crecido más de un 20% y mueve unos 60 millones de Euros en nuestro país.¹

La idea para este proyecto surge cuando comencé a jugar y me di cuenta de que se seguía usando papel y boli para la creación y la gestión de los equipos. Además pienso que en un futuro esta app puede ser la base de un gestor de ligas para la asociación Sky Riders, una de las asociaciones encargada de realizar ligas, no solo de este juego si no también de muchos otros, en nuestra ciudad.

Ya realice un pequeño proyecto con esta idea, pero este surge completamente desde 0, utilizando nuevas herramientas y tecnologías para su realización e implementación.

Actualmente existe una app web llamada tour play². Se trata de un gestor de ligas y de equipos, similar a mi idea pero con muchísimas funcionalidades y muy avanzada. Cuenta con más de 6000 usuarios, 700 competiciones y 800 equipos creados. En ella puedes registrarte en cualquier competición de todo el mundo. Aparte de la gestión de ligas y equipos, ofrece actas y resultados del partido en tiempo real pudiendo anotar cualquier acción mientras se desarrolla el juego. Sin duda, se trata de una app muy bien planteada a la que cada día se unen más usuarios.

Objetivo

El objetivo del proyecto, como ya he comentado, es la de desarrollar una aplicación web que facilite al usuario la gestión y creación de plantillas para sus equipos. Esto agilizaría mucho las cosas, ahorrando tiempo, y ayudaría al jugador a tener todo más organizado, pudiendo consultar la información en cualquier dispositivo y en cualquier lugar con acceso a la red, sin estar pendiente de tener que llevar encima la hoja de plantilla impresa en un papel. Otro de sus objetivos, es el de ayudar al jugador novato a iniciarse, facilitando información del reglamento, lo necesario para jugar y donde poder adquirirlo. En definitiva, hacer más cercano el juego a cualquier usuario, ya sea veterano o acaba de iniciarse en el.

¹ <https://www.rtve.es/noticias/20220403/repor-juegos-mesa/2324338.shtml>

² <https://tourplay.net/es/>

Objective

The aim of the project, as I have already mentioned, is to develop a web application that will make it easier for users to manage and create templates for their teams. This would speed things up a lot, saving time, and would help the player to have everything more organised, being able to consult the information on any device and anywhere with access to the network, without having to carry around a printed sheet of paper. Another of its objectives is to help the novice player to get started, providing information on the rules, what is necessary to play and where to buy it. In short, to bring the game closer to any user, whether they are veterans or new to the game.

2. Acuerdo del proyecto

Requisitos Funcionales y no Funcionales

En este punto se detallan los requisitos que tendrá la aplicación "PlayBloodBow".

Mediante el análisis de requisitos, se podrá identificar las necesidades y restricciones sobre las cuales debe operar la aplicación web. Se trata de un proceso que comienza con el análisis del problema y que finaliza con la documentación de resultados. Es importante identificar todos los requisitos que la aplicación debe cumplir para satisfacer las necesidades de los usuarios finales y de los clientes. Estos requisitos han ido surgiendo bajo mi experiencia tras jugar al juego.

Para el desarrollo de este proyecto, se clasificaron los requisitos de la siguiente forma:

- Los requisitos funcionales determinan el comportamiento de la aplicación, así como sus componentes y funciones a realizar.
- Los requisitos no funcionales definen las características que ha de cumplir la aplicación, como usabilidad, almacenamiento, rendimiento, accesibilidad, etc.

Requisitos Funcionales:

Requisito	RF1 Requisitos generales
Descripción	<ul style="list-style-type: none">• El sistema tiene que mostrar un mensaje de error si una acción del usuario ha salido mal.• El sistema tiene que mostrar un mensaje en caso de éxito de la acción del usuario.• Tiempo de respuesta mínimo.
Tipo	Funcional
Prioridad	Baja
Versión Alta	22/04/2022

Requisito	RF2 Usuarios Registrados
Descripción	Los usuarios se podrán registrar mediante un formulario de registro (email y contraseña) o a través de su cuenta Google.
Tipo	Funcional
Prioridad	Alta
Versión Alta	22/04/2022

Requisito	RF3 Identificación
Descripción	Si el usuario registrado ha iniciado sesión previamente, el inicio será automático.
Tipo	Funcional
Prioridad	Baja
Versión Alta	22/04/2022

Requisito	RF4 Cerrar Sesión
Descripción	El usuario podrá cerrar sesión mediante un botón y será redirigido al login.
Tipo	Funcional
Prioridad	Alta
Versión Alta	22/04/2022

Requisito	RF5 Consultar equipos
Descripción	Los usuarios (anónimos o identificados) podrán consultar equipos por defecto. Los identificados podrán consultar sus propios equipos en otro apartado.
Tipo	Funcional
Prioridad	Alta
Versión Alta	22/04/2022

Requisito	RF6 Consultar tiendas
Descripción	Los usuarios (anónimos o identificados) podrán consultar tiendas.
Tipo	Funcional
Prioridad	Media
Versión Alta	22/04/2022

Requisito	RF7 Consultar reglamento
Descripción	Los usuarios (anónimos o identificados) podrán consultar el reglamento y descargarlo.
Tipo	Funcional
Prioridad	Media
Versión Alta	22/04/2022

Requisito	RF8 Creación y gestión de equipos
Descripción	<p>Los usuarios identificados podrán crear y gestionar sus equipos. Tendrán que rellenar un formulario con los siguientes campos (los 3 primeros son obligatorios):</p> <ul style="list-style-type: none"> • Nombre: Nombre del equipo. • Descripción: Descripción del equipo. • Raza: Raza del equipo. • Imagen: Un link a una imagen que represente a su equipo. • Escudo: Un link a icono que represente a su equipo.
Tipo	Funcional
Prioridad	Alta
Versión Alta	22/04/2022

Requisito	RF9 Añadir jugadores
Descripción	Los usuarios identificados podrán añadir jugadores a sus equipos.
Tipo	Funcional
Prioridad	Alta
Versión Alta	22/04/2022

Requisito	RF10 Eliminar equipos
Descripción	Los usuarios identificados podrán eliminar sus equipos.
Tipo	Funcional
Prioridad	Alta
Versión Alta	22/04/2022

Requisito	RF11 Modificar equipos
Descripción	Los usuarios identificados podrán modificar sus equipos. Solo podrán modificar los campos de descripción , imagen , y escudo .
Tipo	Funcional
Prioridad	Alta
Versión Alta	22/04/2022

** No se descarta añadir más funcionalidades en futuras versiones de la aplicación.*

Requisitos No Funcionales:

Requisito	RNF1 Compatibilidad con navegadores
Descripción	La aplicación web tiene que ser compatible con las últimas versiones de los principales navegadores de Internet (Google Chrome, Mozilla Firefox, Opera...)
Tipo	No Funcional
Prioridad	Baja
Versión Alta	22/04/2022

Requisito	RNF2 Navegación estructurada
Descripción	La navegación por la aplicación web tiene que ser estructurada, atendiendo al árbol de contenidos.
Tipo	No Funcional
Prioridad	Baja
Versión Alta	22/04/2022

Requisito	RNF3 Diseño responsive
Descripción	La interfaz de la aplicación web tiene que ser adaptada a dispositivos móviles, tablets, smartphones.
Tipo	No Funcional
Prioridad	Baja
Versión Alta	22/04/2022

Requisito	RNF4 Diseño
Descripción	La aplicación tiene que tener una interfaz limpia e intuitiva para el usuario, de manera que sea fácil de manejar.
Tipo	No Funcional
Prioridad	Baja
Versión Alta	22/04/2022

Requisito	RNF5 Requisitos LOPD
Descripción	El acceso a los datos de carácter personal se realizará conforme a lo dispuesto en la Ley 15/1999 sobre Protección de Datos de Carácter Personal.
Tipo	No Funcional
Prioridad	Baja
Versión Alta	22/04/2022

Requisito	RNF6 Rendimiento
Descripción	El acceso a la base de datos no supone una carga para la ejecución de la aplicación, por lo que el rendimiento tiene que ser óptimo.
Tipo	No Funcional
Prioridad	Baja
Versión Alta	22/04/2022

Requisito	RNF7 Base de Datos en tiempo real
Descripción	Para aprovechar toda la capacidad de Angular se usará una Base de Datos en tiempo real (Firebase).
Tipo	No Funcional
Prioridad	Baja
Versión Alta	22/04/2022

Tareas

Después de identificar los requisitos funcionales y no funcionales se va a definir las tareas a realizar a lo largo del proyecto. Estas tareas estarán compuestas por sub-tareas como podemos observar en la siguiente tabla. Además se han fijado unas fechas y una duración por cada una de ellas.

ID ↑ ↓	Name	Start Date	End Date	Duration
1	Planificacion	Apr 22, 2022	Apr 26, 2022	3 days
2	▼ Documentacion	Apr 28, 2022	Jun 02, 2022	26 days
3	Diseño BBDD	Apr 28, 2022	May 05, 2022	6 days
4	Diseño Mockup	May 10, 2022	Jun 02, 2022	18 days
5	Documento de Analisis	Apr 28, 2022	Jun 02, 2022	26 days
6	▼ Desarrollo	May 02, 2022	Jun 02, 2022	24 days
7	Creacion estructura del proyecto y compone...	May 02, 2022	May 09, 2022	6 days
8	Implantacion BBDD	May 10, 2022	May 18, 2022	7 days
9	Creacion de Servicios	May 05, 2022	May 12, 2022	6 days
10	Creacion de modelos	May 13, 2022	May 17, 2022	3 days
11	Desarrollo CRUD	May 16, 2022	May 31, 2022	12 days
12	Maquetacion y diseño	May 20, 2022	Jun 02, 2022	10 days
13	Tests	Jun 01, 2022	Jun 02, 2022	2 days

Metodología

Para la realización de este proyecto he optado el uso de metodologías Ágiles, concretamente la metodología SCRUM.

Las metodologías ágiles son aquellas que permiten adaptar la forma de trabajo a las condiciones del proyecto, consiguiendo flexibilidad e inmediatez en la respuesta para amoldar el proyecto y su desarrollo a las circunstancias específicas del entorno.

Entre sus ventajas encontramos:

- Mejora de la calidad del producto
- Mayor satisfacción del cliente
- Trabajo colaborativo
- Uso de métricas más relevantes
- Mayor control y capacidad de predicción
- Reducción de costes

En concreto la metodología SCRUM se caracteriza por ser la “metodología del caos” que se basa en una estructura de desarrollo incremental, esto es, cualquier ciclo de desarrollo del producto y/o servicio se desgrana en pequeñas tareas divididas en distintas etapas: análisis, desarrollo y testing. En la etapa de desarrollo encontramos lo que se conoce como interacciones del proceso o Sprint.



Esta metodología pasa por diferentes fases que hacen posible que se lleve a cabo con éxito. Estas fases son:

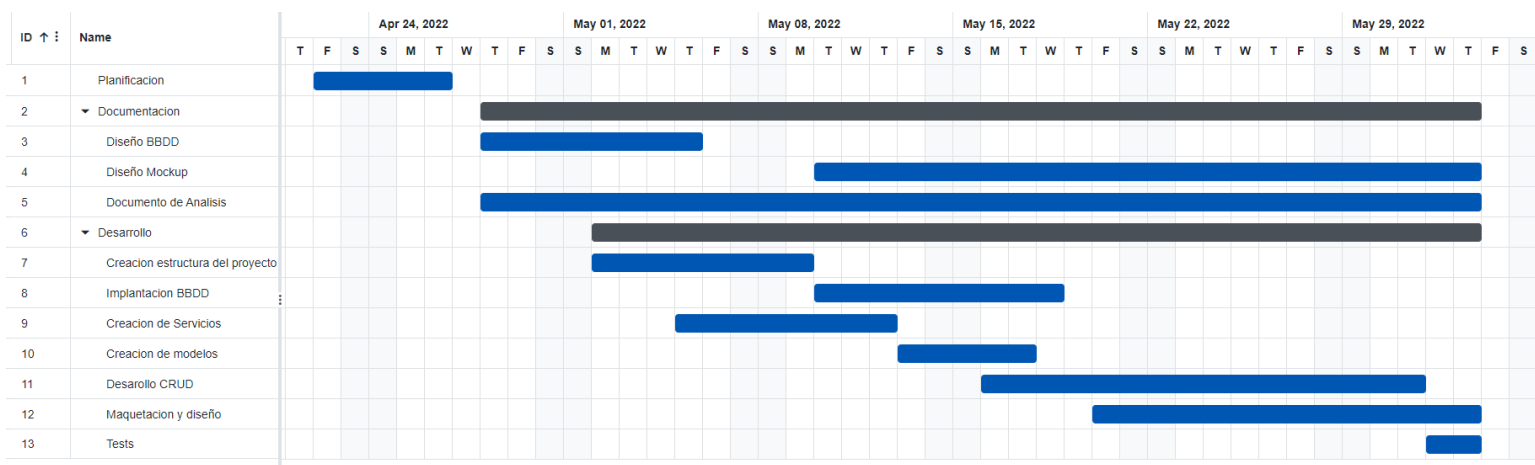
- **Planificación: Product Backlog:** Esta es la fase en la que se establecen las tareas prioritarias y donde se obtiene información breve y detallada sobre el proyecto que se va a desarrollar. Con el método Scrum no es necesario definir todos los objetivos al comienzo del proyecto.
- **Ejecución: Sprint:** Dentro del método Scrum, el Sprint es el corazón, un intervalo de tiempo que como máximo tiene una duración de un mes y en donde se produce el desarrollo de un producto que es entregable potencialmente. También se puede definir el Sprint como un mini proyecto en donde el equipo de trabajo se focaliza en el desarrollo de tareas para alcanzar el objetivo que se ha definido previamente en el Sprint planning.
- **Control: Burn Down:** Es la fase en la que se mide el progreso de un determinado proyecto Scrum. En ella, el Scrum Master será el encargado de actualizar los gráficos cuando se finalice cada uno de los Sprint.

En conclusión, la metodología Scrum es una metodología ágil que hace énfasis en el trabajo en equipo donde la claridad de los objetivos es crucial para avanzar hacia una versión cada vez mejor.

Planificación temporal de las tareas

Tras analizar las tareas a realizar, el proyecto tiene una dedicación de unas 20h semanales de lunes a viernes, unas 4h-5h al día.

Como se puede observar en el siguiente diagrama de Gantt, la planificación está estructurada para conseguir finalizar el proyecto en el plazo estimado.



Durante el Product Backlog se llevó a cabo la fase de planificación, donde se definen las bases sobre las que se va a trabajar durante todo el proceso. Primero se lleva a cabo una investigación del mercado, la competencia y los usuarios. Estos datos obtenidos se utilizarán para crear los conceptos iniciales que servirán de ayuda para dar forma a los contenidos.

En el primer y segundo Sprint, se definen los aspectos del diseño de la aplicación mediante la elaboración de la base de datos y mockups.

El tercer Sprint, durante la fase de desarrollo, se programa la aplicación y se implementan todas las funcionalidades que harán que el proyecto funcione. Para finalizar, durante el cuarto, y último Sprint, se realizan los tests pertinentes para comprobar el correcto funcionamiento de esta.

Presupuesto

	Costes fijos mensuales
Suministros	70€
Internet, teléfono	35€
Salario neto	1050€
Seguridad Social e IRPF	239€
Cuota nuevos autónomos	60€
Ordenador portátil	$800/12 = 67€$
Total	1521€

Como podemos observar los gastos mensuales del proyecto son en torno a 1521€. Pese a haber desarrollado el proyecto en un mes más o menos, por si acaso contaremos con estos gastos para dos meses, por si surgieran problemas o cualquier otro imprevisto.

Así pues los gastos totales para la realización de este proyecto son: **$1521 \times 2 = 3042€$**

3. Análisis y diseño

En este apartado encontraremos los elementos UML como diagrama de Entidad-Relación, de clases, o casos de usos para entender el funcionamiento de PlayBloodBowl y como está estructurada la información.

También se muestran los mockups de diseño de la interfaz y finalmente hablaremos de las tecnologías usadas y de la arquitectura de los componentes

Diagrama E-R

Pese a que Firebase es una Base de Datos no relacional, si que encontramos en cierta manera elementos de BBDD relacionales como por ejemplo la propagación de claves para hacer referencia a otras colecciones/documentos. En los siguientes diagramas podemos ver los componentes junto con sus atributos y la manera en que están relacionados.

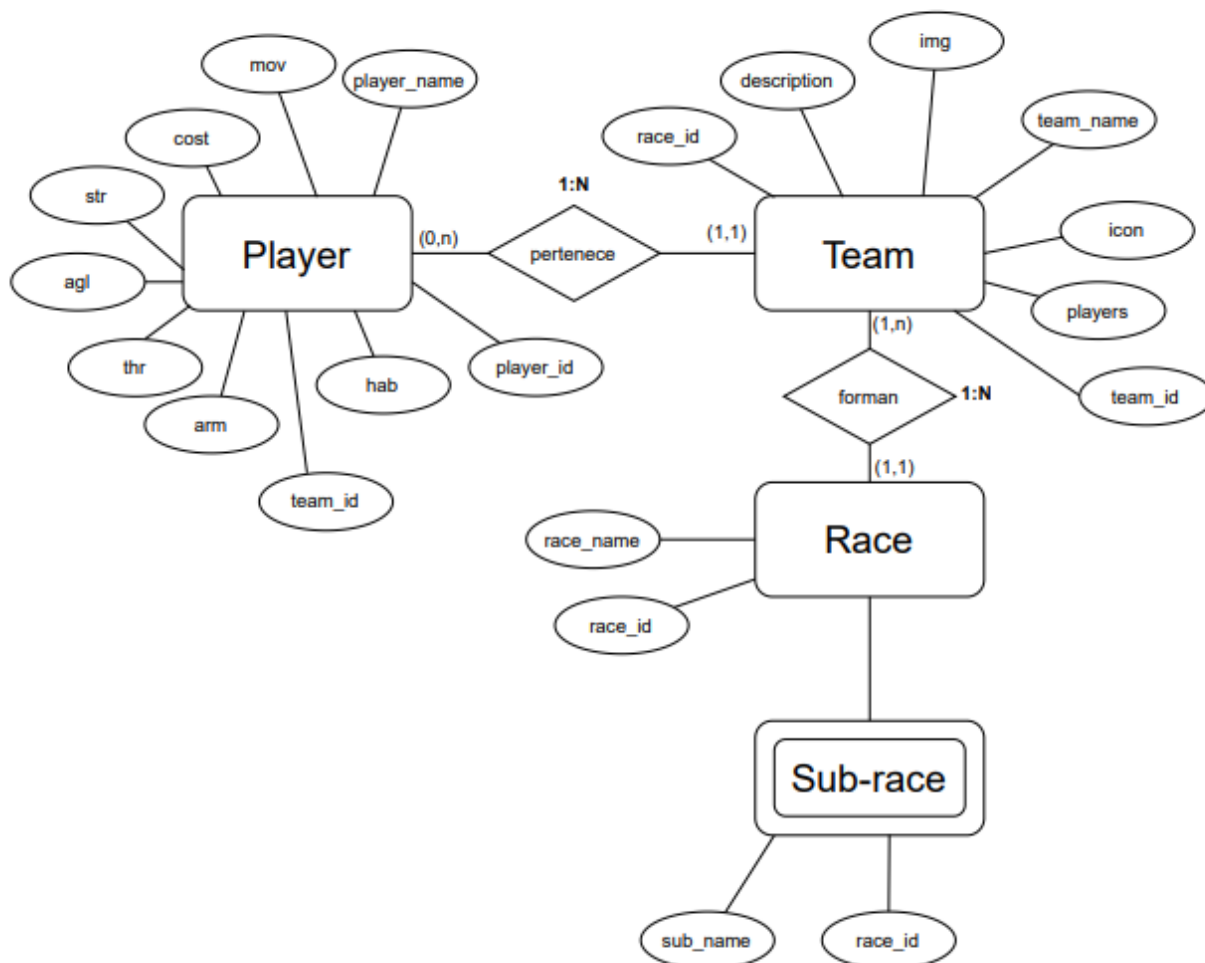
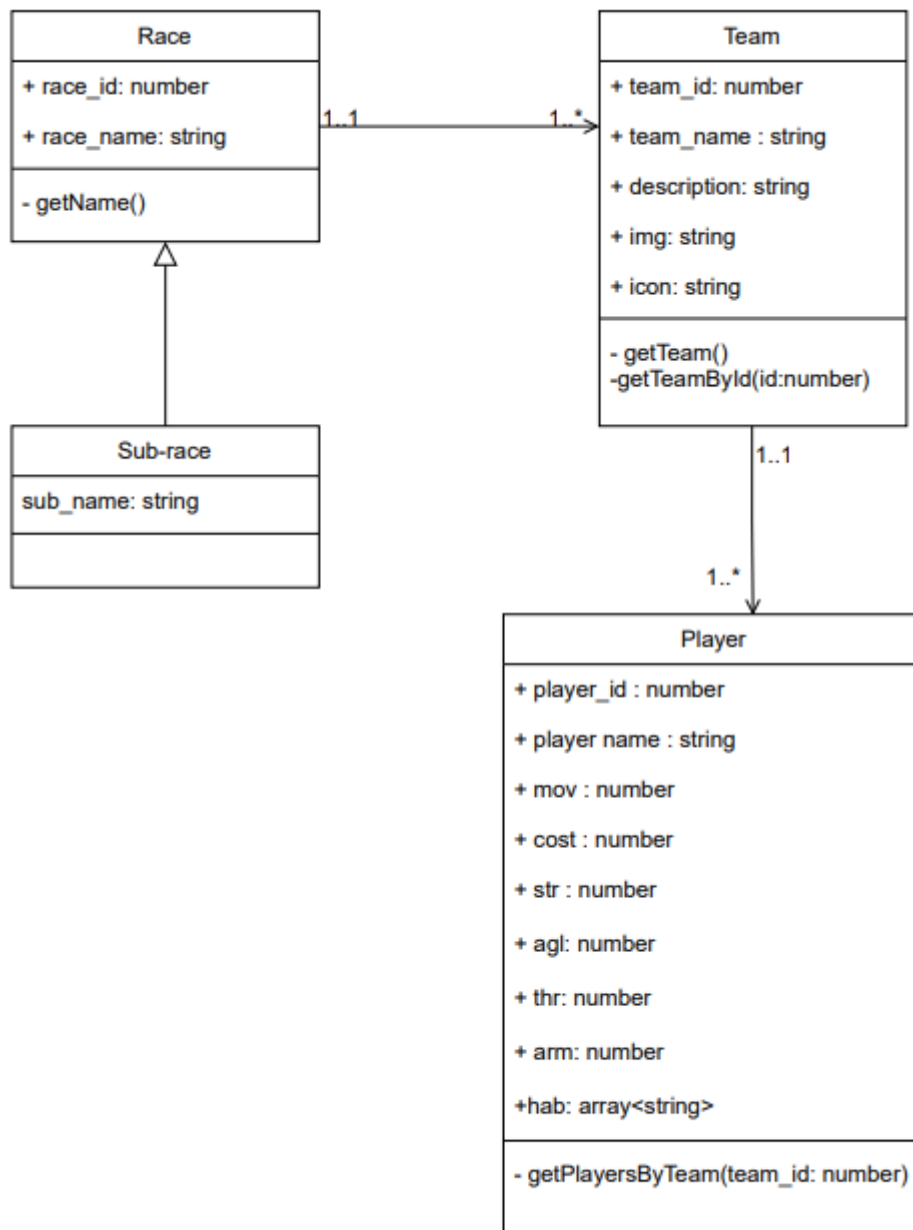
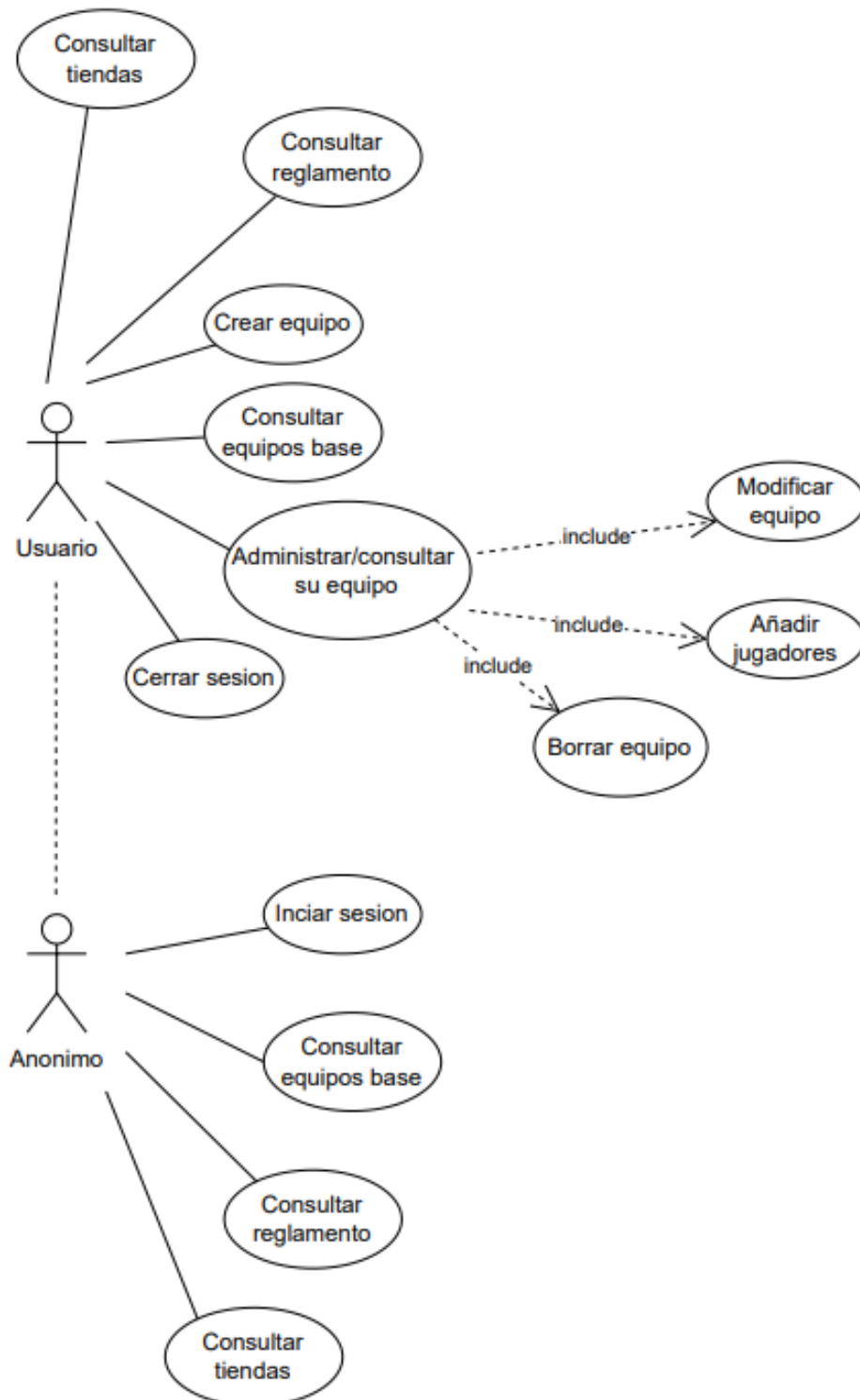


Diagrama de clases



Casos de Uso

A continuación veremos los casos de uso, un total de 10, en el que participan dos actores (Anónimo y Usuario). Dependiendo de si el usuario está identificado o no, podrá acceder a determinadas funciones que un usuario anónimo no podrá hacer.



Caso de Uso 1	Iniciar sesión/Registrarse
Descripción	Un usuario inicia sesión/se registra en el sistema.
Actores	Anónimo.
Precondición	Un anónimo quiere registrarse o iniciar sesión en el sistema.
Postcondición	El anónimo queda identificado en el sistema y pasa a ser usuario del sistema.
Flujo normal	<ol style="list-style-type: none"> 1. Un anónimo quiere registrarse/iniciar sesión en el sistema. 2. El sistema solicita su usuario y contraseña. 3. El anónimo introduce los datos. 4. El sistema valida los datos. 5. El sistema identifica al usuario y crea una sesión.
Flujo alternativo	Si los datos introducidos por el usuario son incorrectos se muestra un error y no se inicia sesión.

Caso de Uso 2	Consultar equipos
Descripción	Un usuario quiere consultar equipos.
Actores	Anónimo y usuario identificado.
Precondición	El usuario entra en la opción de consultar equipos.
Postcondición	El usuario puede ver el listado de los equipos por defecto.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario quiere consultar los equipos. 2. El sistema carga la lista de los equipos por defecto y los muestra
Flujo alternativo	

Caso de Uso 3	Consultar Reglamento
Descripción	Un usuario quiere consultar el reglamento
Actores	Anónimo y usuario identificado.
Precondición	El usuario entra en la opción de consultar el reglamento.
Postcondición	El usuario puede descargar el reglamento.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario entra en la quiere consultar el reglamento y accede a la opción de Reglamento. 2. El usuario pincha en el botón de descargar reglamento. 3. El sistema le redirecciona a una página externa donde puede consultar y descargar el pdf con las reglas.
Flujo alternativo	

Caso de Uso 4	Consultar tiendas
Descripción	Un usuario quiere consultar tiendas.
Actores	Anónimo y usuario identificado.
Precondición	El usuario entra en la opción de consultar tiendas.
Postcondición	El usuario puede ver el listado de tiendas en Zaragoza.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario quiere consultar tiendas y pincha en Tiendas. 2. El sistema muestra unas tarjetas con información de tiendas.
Flujo alternativo	

Caso de Uso 5	Crear equipos
Descripción	Un usuario quiere crear un equipo.
Actores	Usuario identificado.
Precondición	<ol style="list-style-type: none"> 1. El usuario tiene que estar identificado. 2. El usuario entra en la opción de añadir equipos.
Postcondición	El usuario ha creado un equipo.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario quiere crear su equipo. 2. El sistema carga un formulario con distintos campos para crear el equipo. 3. El sistema guarda la información del nuevo equipo.
Flujo alternativo	

Caso de Uso 6	Administrar/Consultar equipo
Descripción	Un usuario quiere administrar o consultar su equipo.
Actores	Usuario identificado.
Precondición	<ol style="list-style-type: none"> 1. El usuario tiene que estar identificado. 2. El usuario entra en la opción de administrar equipos.
Postcondición	El usuario consulta o administra su equipo.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario quiere administrar su equipo. 2. El sistema carga un listado de sus equipos. 3. El usuario selecciona el equipo que quiere administrar y el sistema muestra la info de ese equipo.
Flujo alternativo	Si el usuario todavía no ha creado equipos el sistema no mostrará ningún equipo.

Caso de Uso 7	Añadir jugadores a su equipo
Descripción	Un usuario quiere añadir jugadores a su equipo.
Actores	Usuario identificado.
Precondición	<ol style="list-style-type: none"> 1. El usuario tiene que estar identificado. 2. El usuario entra en la opción de administrar equipos. 3. El usuario selecciona el equipo que quiere administrar
Postcondición	El usuario puede añadir/ha añadido jugadores.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario quiere añadir jugadores a su equipo. 2. El usuario accede a administrar equipos. 3. El sistema carga un listado de sus equipos. 4. El usuario selecciona el equipo que quiere administrar y el sistema muestra la info de ese equipo. 5. El sistema muestra un desplegable con los jugadores disponibles para la raza del equipo del usuario. 6. El usuario añade los jugadores seleccionados. 7. El sistema guarda esta información.
Flujo alternativo	Si el usuario todavía no ha creado equipos el sistema no mostrará ningún equipo.

Caso de Uso 8	Modificar su equipo
Descripción	Un usuario quiere modificar su equipo.
Actores	Usuario identificado.
Precondición	<ol style="list-style-type: none"> 1. El usuario tiene que estar identificado. 2. El usuario entra en la opción de administrar equipos. 3. El usuario selecciona el equipo que quiere administrar
Postcondición	El usuario puede modificar/ha modificado su equipo.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario quiere modificar su equipo. 2. El usuario accede a administrar equipos. 3. El sistema carga un listado de sus equipos. 4. El usuario selecciona el equipo que quiere administrar y el sistema muestra la info de ese equipo. 5. El usuario pincha en la opción de editar equipo. 6. El sistema muestra un formulario y permite al usuario modificar algunos campos. 7. El usuario guarda los campos modificados y el sistema registra estos cambios..
Flujo alternativo	Si el usuario todavía no ha creado equipos el sistema no mostrará ningún equipo.

Caso de Uso 9	Eliminar equipo
Descripción	Un usuario quiere eliminar su equipo.
Actores	Usuario identificado.
Precondición	<ol style="list-style-type: none"> 1. El usuario tiene que estar identificado. 2. El usuario entra en la opción de administrar equipos.
Postcondición	El usuario ha borrado su equipo.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario quiere eliminar su equipo. 2. El usuario accede a administrar equipos. 3. El sistema carga un listado de sus equipos. 4. El usuario selecciona el equipo que quiere administrar y el sistema muestra la info de ese equipo. 5. El usuario pincha en la opción de eliminar. 6. El sistema elimina esa información.
Flujo alternativo	Si el usuario todavía no ha creado equipos el sistema no mostrará ningún equipo.

Caso de Uso 10	Cerrar sesión
Descripción	Un usuario quiere cerrar sesión.
Actores	Usuario identificado.
Precondición	<ol style="list-style-type: none"> 1. El usuario tiene que estar identificado. 2. El usuario pincha en logout.
Postcondición	El usuario ha cerrado sesión.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario quiere cerrar sesión. 2. El usuario hace click en la opción de cerrar sesión. 3. El sistema cierra la sesión de ese usuario. 4. El sistema lo redirecciona al login.
Flujo alternativo	

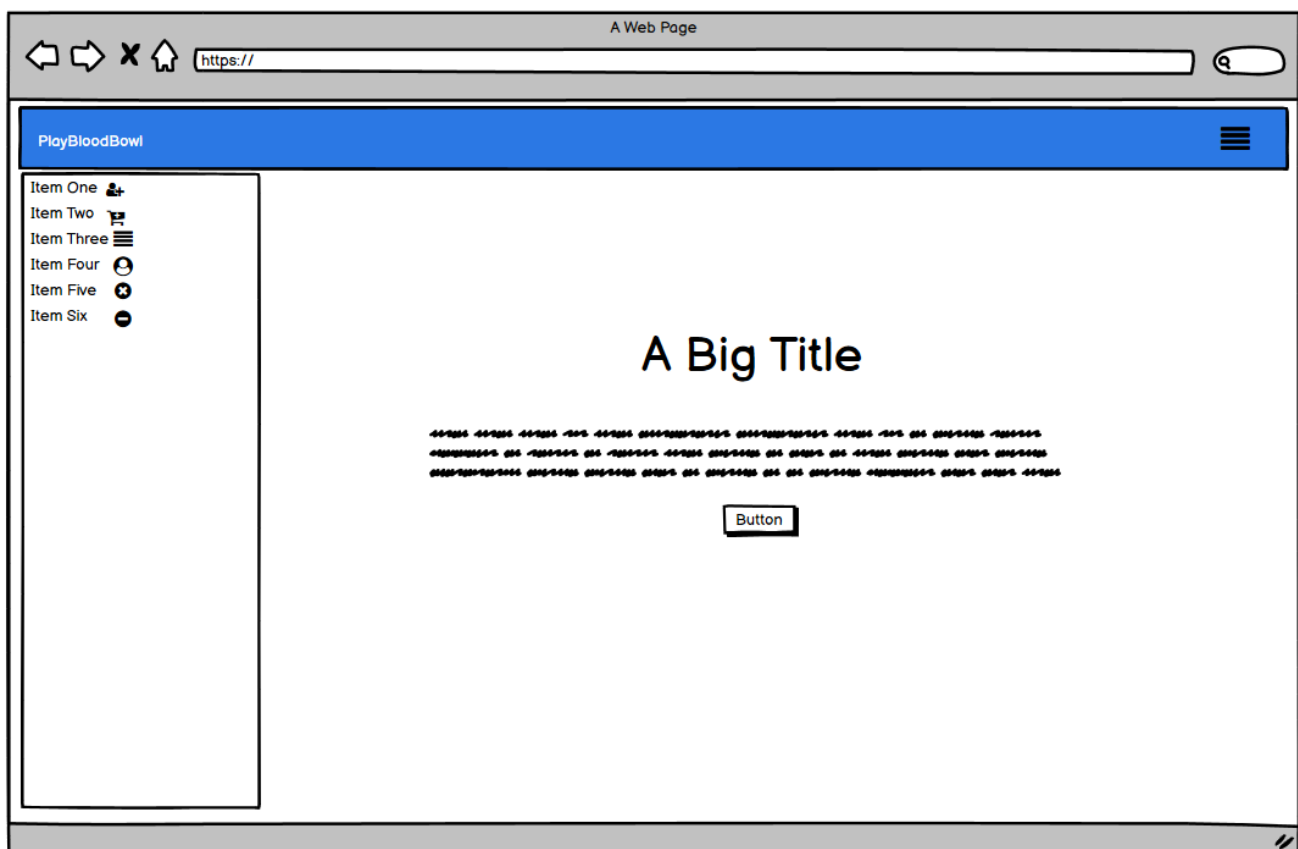
Mockups

Para el diseño de la aplicación he optado por una interfaz limpia, que facilite la navegación entre las distintas opciones que nos ofrece, poco recargada de imágenes, con fondos blancos y toques del color **#384bb4**, utilizado para las tarjetas o en el navbar. Para los textos el color es negro o blanco dependiendo del color del fondo donde se encuentren.

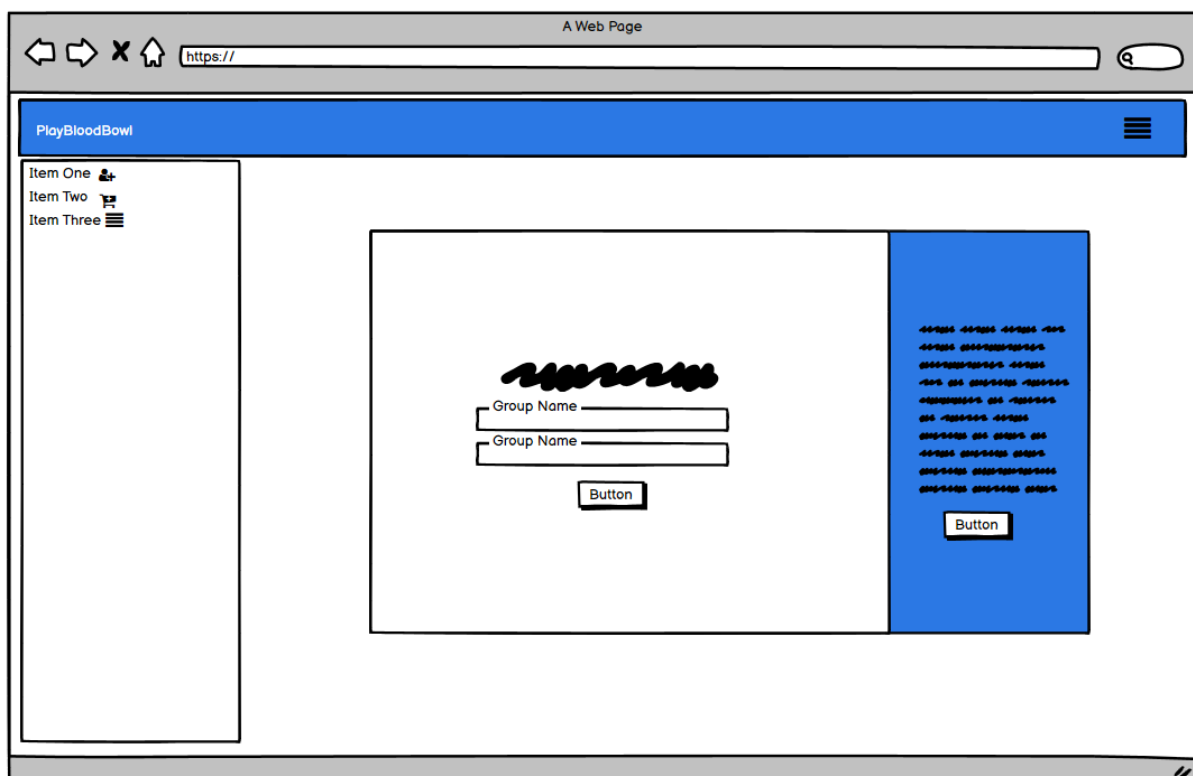
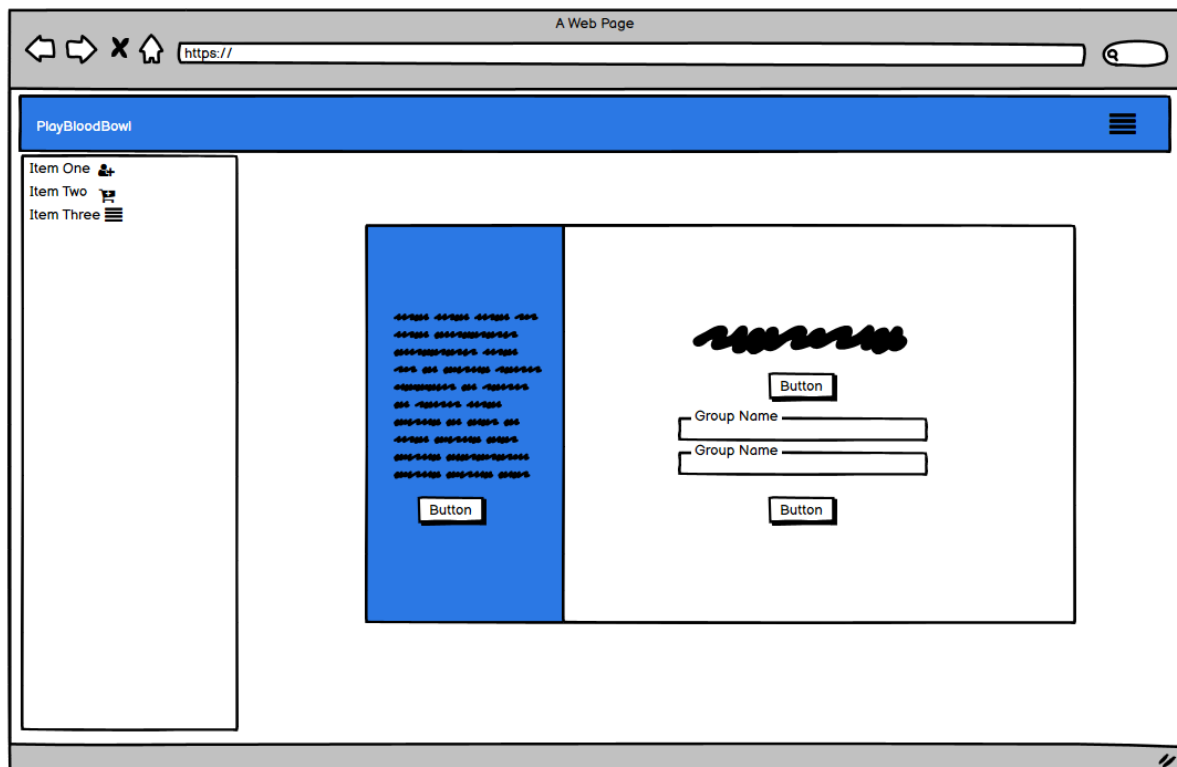
La fuente utilizada es “Roboto”, de la familia de fuentes tipográficas del tipo “sans-serif” desarrollada por Google como fuente del sistema operativo móvil Android 4.0 “Ice Cream Sandwich”. Google describe la fuente como “moderna, pero a la vez accesible” y “emocional”.

No se descartan cambios en el diseño de la aplicación ya que en estos spints el desarrollo se ha centrado más en que sea un app completamente funcional, dejando en segundo plano la interfaz, aunque siempre intentando que sea lo más amigable y limpia posible.

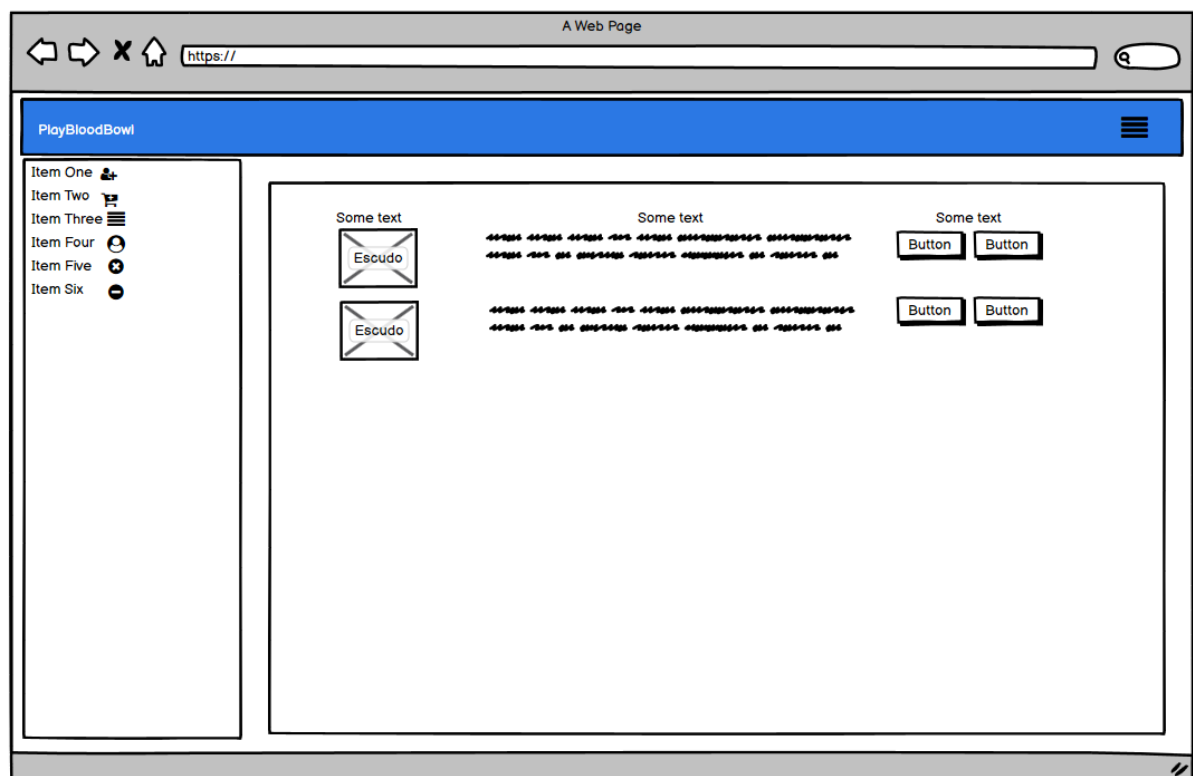
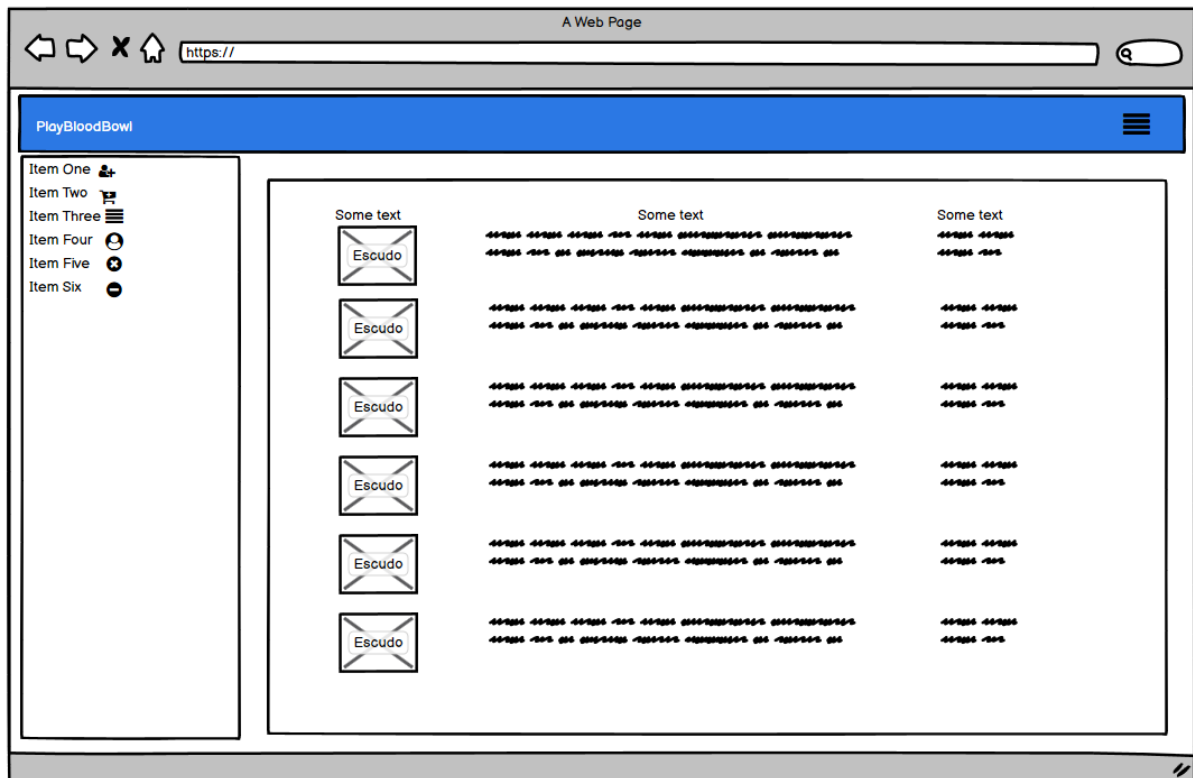
Página de inicio:



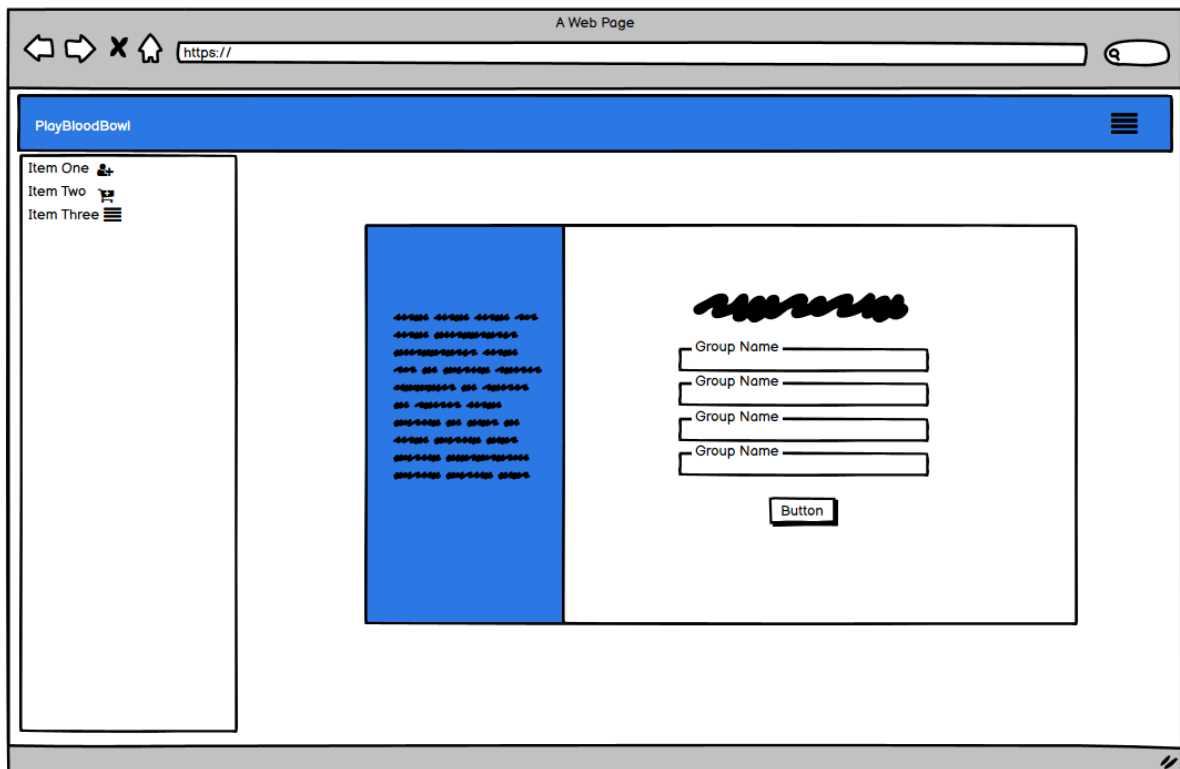
Registro y login:



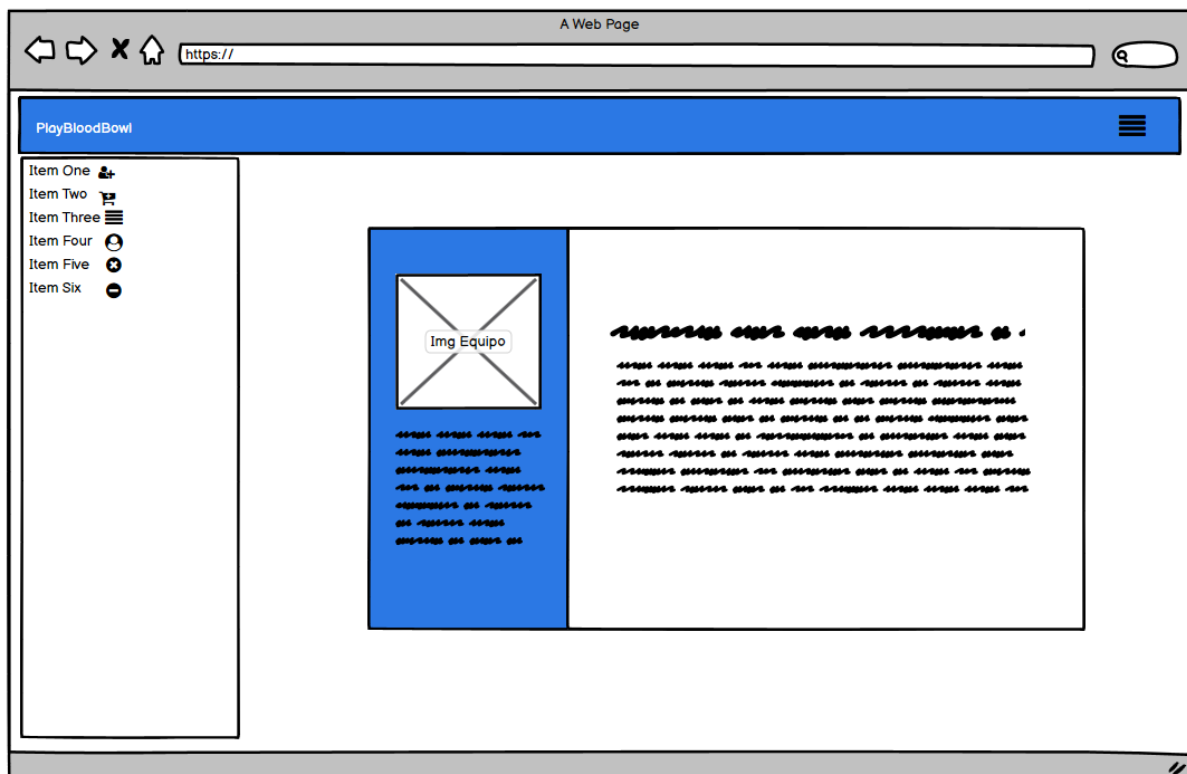
Listas de equipo:

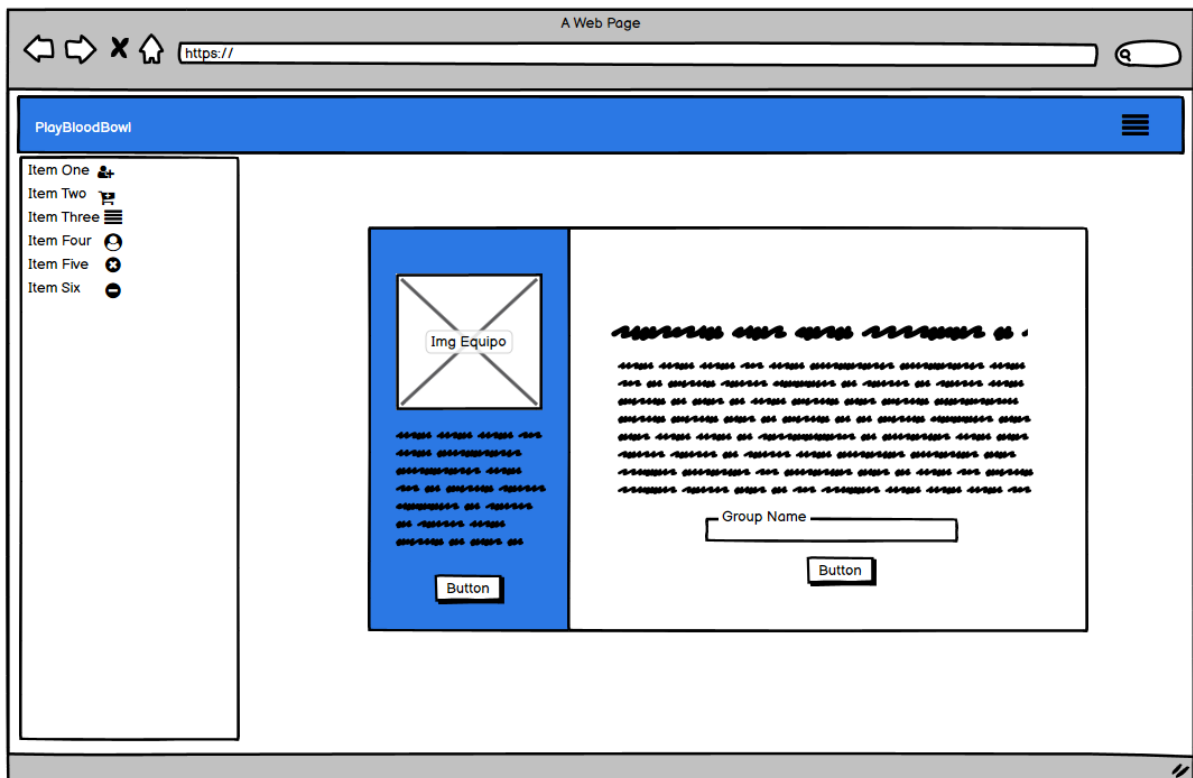


Añadir y modificar equipos:



Detalle de equipos:





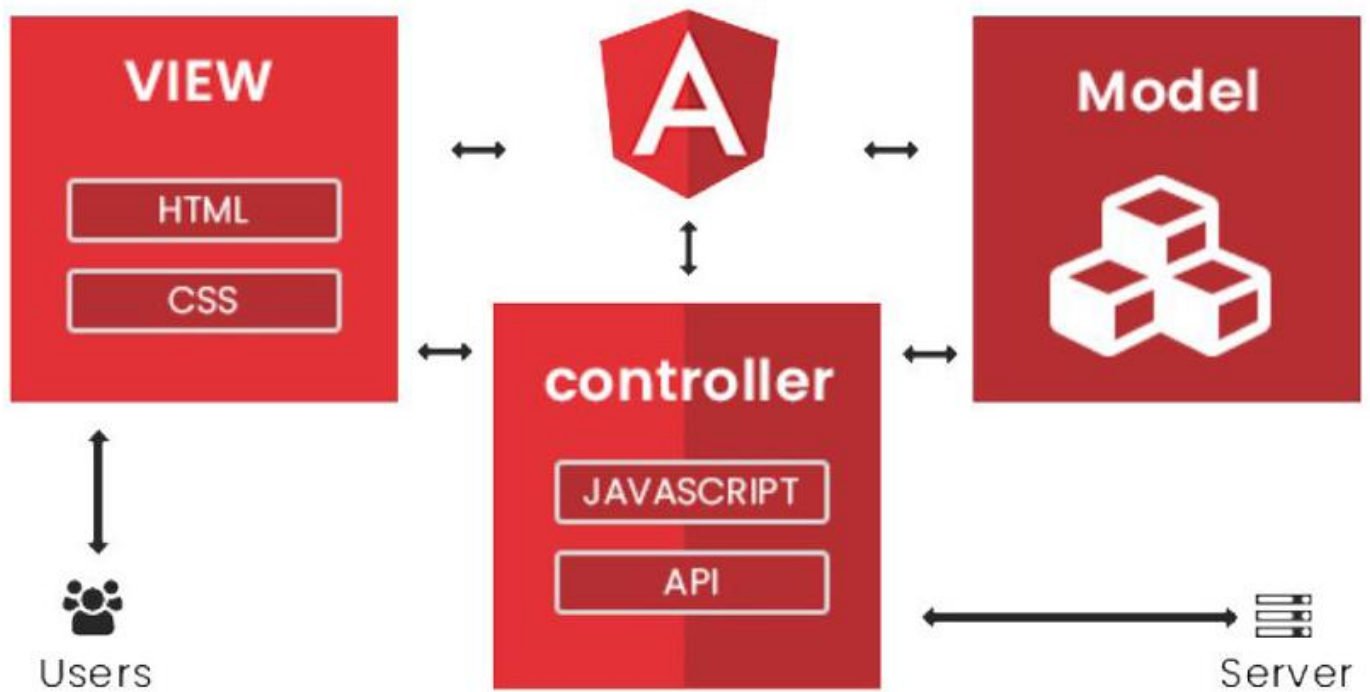
Tecnologías/Herramientas usadas

En un principio para el desarrollo del proyecto iba a utilizar Python y Django, mysql para la gestión de la base de datos y css junto con bootstrap para los estilos. Mi idea era continuar con un proyecto ya iniciado con estas tecnologías, añadiendo funcionalidades que no estaban implementadas pero que son muy necesarias para desarrollar totalmente la idea y para hacer una app funcional. Finalmente, deseché el seguir con este proyecto y empezar uno de cero usando nuevas tecnologías y herramientas.

El cambio de idea respecto al uso de otras tecnologías viene cuando estando ya en la empresa realizando las prácticas se me propuso utilizar angular (el framework que voy a usar principalmente en el proyecto al que estoy asignando). Por lo que utilizar para el desarrollo de mi app este framework es una forma excelente de aprender Angular e irme familiarizando con él.

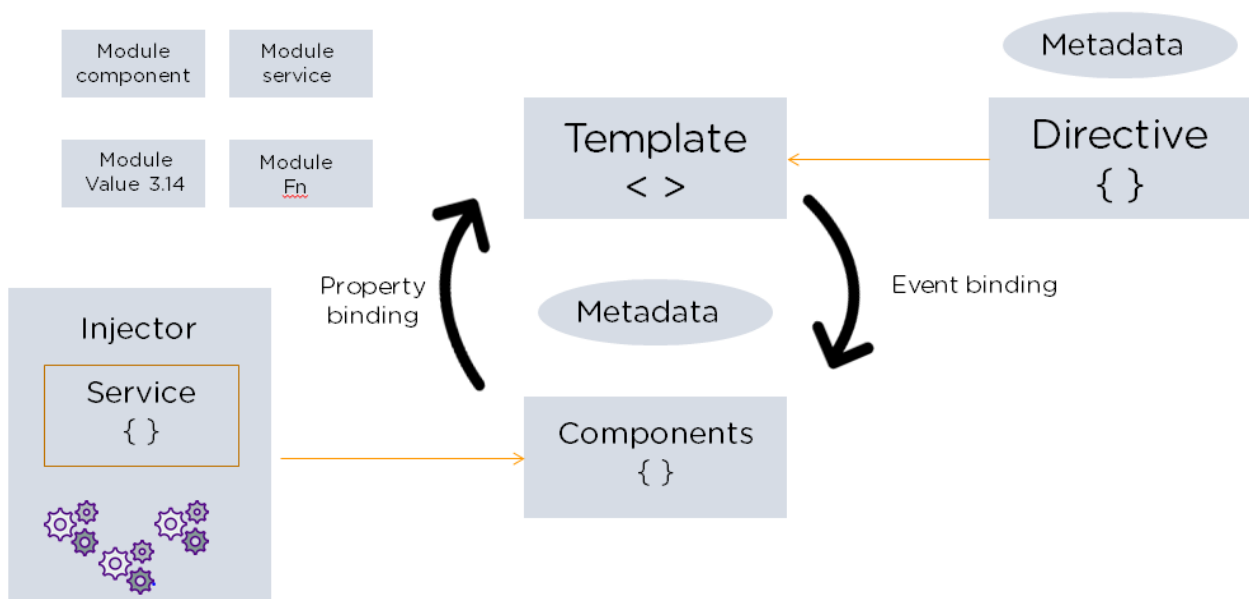
Angular es un Framework de JavaScript de código abierto escrito en TypeScript. Su objetivo principal es desarrollar aplicaciones de una sola página. Google se encarga del mantenimiento y constantes actualizaciones de mejoras para este framework.

Es un marco modelo-vista-controlador (MVC) completo. Proporciona una guía clara sobre cómo se debe estructurar la aplicación y ofrece un flujo de datos bidireccional al tiempo que proporciona un DOM real.



Para construir aplicaciones Angular creamos:

- Templates HTML que contienen etiquetas especiales de Angular.
- Componentes de clase que gestionan dichos templates.
- Servicios que encapsulan la lógica de la app.
- Módulos que organizan estos componentes y servicios.



La gran mayoría de los módulos de Angular utilizan una biblioteca muy potente llamada RxJS. Es una biblioteca para programación reactiva basada en el patrón de “observador”. Angular se apoya en RxJS para habilitar la reacción a eventos de forma asíncrona y con alto desempeño.

Otra de las herramientas claves en el desarrollo con Angular es su CLI (Command line interface). Es una herramienta de línea de comando que facilita la creación de aplicaciones creando la estructura base con archivos necesarios y esqueleto de código, organizándolos en una estructura de directorios estandarizada y agregando las bibliotecas y dependencias requeridas. También soporta comandos para ejecutar pruebas y correr la app, entre muchos otros.

Para los estilos he utilizado Angular Material, una librería de estilos basada en la guía de diseño de Material Design, realizado por el equipo de Angular para integrarse perfectamente con Angular. Junto a Angular Material también utilizo SASS, un framework de CSS. Su principal característica es que convierte los CSS en algo dinámico, permite trabajar más rápido en la creación de código con la posibilidad de crear funciones y reutilizar código gracias a los mixins, variables que nos permiten guardar valores. Se puede usar en ficheros .sass o .scss. Los .scss ya permiten incorporar código de CSS clásico.

Tanto la sintaxis de .sass como la de .scss no puede ser interpretada directamente por los navegadores, por lo que se debe compilar para traducir estos archivos en un fichero CSS.

Finalmente, para la base de datos me he decantado por Firebase, ya que la combinación con Angular es muy potente y muy utilizada en el mundo empresarial.

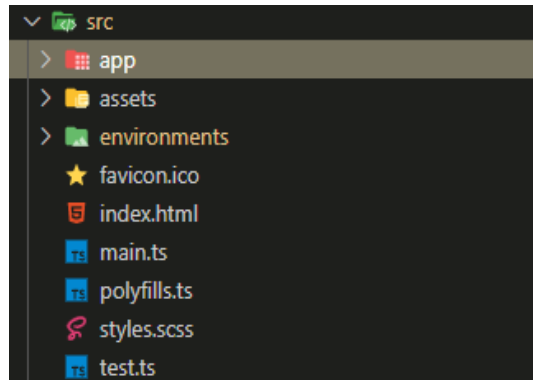
Firebase es una plataforma en la nube desarrollada por Google. Su función esencial es hacer más fácil la creación de app, tanto para web como para móvil. Cuenta con gran cantidad de herramientas.

Una de sus principales características es que ofrece bases de datos en tiempo real. Estas se alojan en la nube, y se tratan de bases de datos no sql, almacenando los datos en JSON. Alojan y disponen de la información en tiempo real manteniéndolos actualizados. Esta funcionalidad hace que la combinación con angular sea tan potente, ya que cualquier cambio en los datos se verá reflejado automáticamente en la app sin tener que recargar o volver a lanzar consultas.

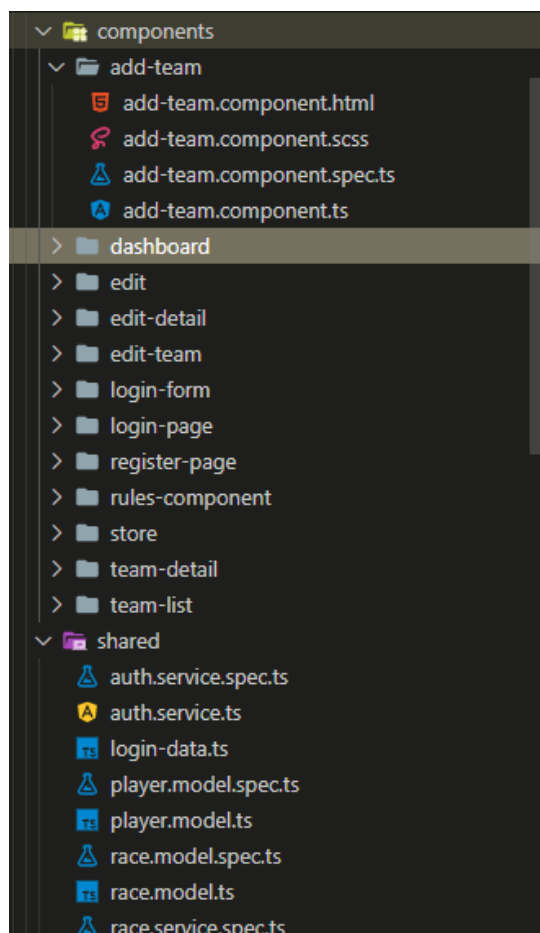
Firebase ofrece un servicio de gestión de usuarios que además permite la identificación de los mismos con diferentes proveedores, por ejemplo Google, Facebook, GitHub... También un método de autenticación básica con usuario y contraseña. Aunque si que es un nivel básico por lo general, para una app como la de este proyecto es más que suficiente.

Arquitectura de componentes

Como ahora veremos, Angular nos ofrece una organización limpia y bien estructurada por carpetas y módulos, muy bien relacionados entre sí. Además, gracias a Angular CLI es muy sencillo el poder generar estos archivos, ya que con un solo comando el propio Angular nos genera y ordena estos componentes.



Dentro de nuestra carpeta raíz (TFG-DAVIDPEREZ-GESTORBB) se encuentran los archivos generados por el propio angular para el funcionamiento de la app. También encontramos la carpeta src que contiene los archivos que van a componer nuestra app. En esta carpeta src, encontramos archivos globales para la app como styles.scss, index.html... Y carpetas como app o enviroments, en la cual conectaremos con nuestra base de datos.



Dentro de app es donde se encuentran los componentes y los módulos que angular genera para toda la lógica de nuestra app.

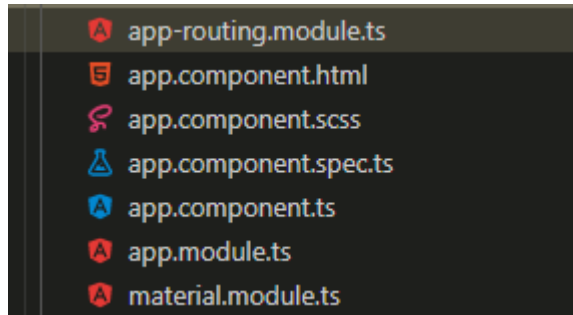
En la carpeta components encontraremos los distintos componentes junto con sus plantillas de estilo, de html y de typeScript.

En shared tendremos los modelos de nuestra aplicación, archivos .ts donde crearemos estas clases Y también el servicio que nos permitirá conectarnos con nuestro backend.

Finalmente encontramos los archivos globales como pueden ser los módulos y los componentes principales de nuestra app. El módulo principal es app.module.ts, donde recogeremos todo lo necesario para el correcto funcionamiento.

En el módulo `app-routing.module.ts` encontraremos las rutas para poder navegar por nuestra aplicación. Angular material estará recogido en el módulo `material.module.ts`.

`App.component.html` será nuestra plantilla html base, donde recogeremos el navbar y el sidenav, siendo global a toda la aplicación, y donde cargaremos el resto de plantillas de los distintos componentes.



4. Documento de implementación, pruebas e implantación

En este apartado trataremos la implementación y las pruebas funcionales para su correcto funcionamiento de la app.

Implementación

Para su implementación hemos necesitado:

```
"@angular/animations": "~13.3.0",
"@angular/cdk": "^13.3.6",
"@angular/common": "~13.3.0",
"@angular/compiler": "~13.3.0",
"@angular/core": "~13.3.0",
"@angular/fire": "^7.3.0",
"@angular/forms": "~13.3.0",
"@angular/material": "^13.3.6",
"@angular/platform-browser": "~13.3.0",
"@angular/platform-browser-dynamic": "~13.3.0",
"@angular/router": "~13.3.0",
"firebase": "^9.8.1",
"rxjs": "~7.5.0",
"tslib": "^2.3.0",
"zone.js": "~0.11.4"
```

Y para el control de versiones se está utilizando el repositorio de **GitHub**:

- https://github.com/DavidPerezVelilla/TFG_DavidPerezVelilla

Pruebas

Para las pruebas Angular nos lo pone fácil. A la hora de crear un componente, también se crean los archivos .spec.ts donde el propio Framework realiza una serie de tests para comprobar el funcionamiento de estos. Se encuentran en la propia carpeta del componente, junto a los archivos .html, .scss y .ts. Para ejecutar estos tests simplemente tenemos que ejecutar el comando “ng test” que iniciará el corredor de pruebas de Karma.

Este es el ejemplo de uno de los test:

```
describe('DashboardComponent', () => {
  let component: DashboardComponent;
  let fixture: ComponentFixture<DashboardComponent>;

  beforeEach(async () => {
    await TestBed.configureTestingModule({
      declarations: [ DashboardComponent ]
    })
      .compileComponents();
  });

  beforeEach(() => {
    fixture = TestBed.createComponent(DashboardComponent);
    component = fixture.componentInstance;
    fixture.detectChanges();
  });

  it('should create', () => {
    expect(component).toBeTruthy();
  });
});
```

Y aquí cuando ejecutamos el comando:

Karma v 6.3.19 - connected; test: complete;

DEBUG

Chrome 102.0.5005.63 (Windows 10) is idle

Pruebas Funcionales:

Requisito	RF1 Requisitos Generales
Descripción	El sistema tiene que mostrar un error si una acción de usuario ha salido mal. El sistema muestra un mensaje en el caso de éxito de una acción.
Datos de entrada	Añadir equipo, añadir jugadores, borrar equipo, editar equipo.
Precondición	El usuario tiene que estar identificado.
Datos de Salida	Si algún campo que sea requerido está vacío se muestra en rojo. OK Si todo ha ido bien, se muestra un mensaje de éxito. OK

Requisito	RF2 Usuarios Registrados
Descripción	Los usuarios registrados pueden acceder a la aplicación web mediante su usuario y contraseña o cuenta Google.
Datos de entrada	Se insertan los datos en el login.
Precondición	El usuario debe tener la cuenta creada, o crear una.
Datos de Salida	Si las credenciales son válidas se le redirige a la página principal. OK Si no son válidas muestra un mensaje de error. OK Si al introducir la contraseña para registrarse es mejor de 6 caracteres muestra un error. OK Si el correo no tiene formato email muestra un error. OK

Requisito	RF3 Identificación
Descripción	Si el usuario registrado ha iniciado sesión y no sale de ella, al volver a entrar la identificación es automática.
Datos de entrada	Se accede a la aplicación.
Precondición	El usuario tiene que estar registrado.
Datos de Salida	El usuario accede automáticamente. OK Si el usuario cierra sesión, el sistema vuelve a pedir los datos. OK

Requisito	RF4 Cerrar Sesión
Descripción	Al cerrar sesión haciendo click en Logout el sistema redirecciona nuevamente al login.
Datos de entrada	Click en Logout.
Precondición	El usuario tiene que estar identificado.
Datos de Salida	El usuario cierra sesión y le redirecciona al login. OK

Requisito	RF5 Menú Lateral
Descripción	Al iniciar sesión en el menú lateral aparecerán opciones que no están disponibles para usuarios anónimos.
Datos de entrada	El usuario accede a la aplicación.
Precondición	El usuario tiene que estar identificado.
Datos de Salida	El usuario identificado visualiza las opciones ocultas. OK El usuario anónimo no las visualiza. OK

Requisito	RF6 Crear Equipos
Descripción	El usuario se dispone a crear nuevos equipos.
Datos de entrada	Se crea un nuevo equipo.
Precondición	El usuario tiene que estar identificado.
Datos de Salida	El usuario rellena el formulario y cuando está correcto se habilita el botón de crear. OK El sistema registra el equipo en la Base de Datos. OK El sistema muestra un mensaje de éxito. OK

Requisito	RF6 Borrar Equipos
Descripción	El usuario se dispone a borrar equipos.
Datos de entrada	Se borra un equipo.
Precondición	El usuario tiene que estar identificado.
Datos de Salida	El usuario hace click en el botón de borrar y el sistema muestra un mensaje de éxito. OK El sistema borra al equipo en la Base de Datos. OK El sistema redirecciona al usuario a la página de inicio. OK

Requisito	RF7 Editar Equipos
Descripción	El usuario se dispone a editar equipos.
Datos de entrada	Se edita equipo.
Precondición	El usuario tiene que estar identificado.
Datos de Salida	<p>El usuario hace click en editar equipo y el sistema le muestra un formulario. OK</p> <p>El usuario rellena rellena el formulario y cuando está correcto se habilita el botón de actualizar. OK</p> <p>El sistema muestra un mensaje de éxito. OK</p> <p>El sistema actualiza el equipo en la Base de Datos. OK</p> <p>El sistema redirecciona al usuario a la página de inicio. OK</p>

Requisito	RF8 Añadir Jugadores
Descripción	El usuario se dispone a añadir jugadores a su equipo.
Datos de entrada	Se añaden jugadores.
Precondición	El usuario tiene que estar identificado.
Datos de Salida	<p>El usuario selecciona el jugador a añadir y clica en añadir jugador. OK</p> <p>El sistema muestra un mensaje de éxito. OK</p> <p>El sistema añade el jugador a la Base de Datos. OK</p> <p>El sistema redirecciona al usuario a la página de inicio. OK</p>

Requisito	RF9 Ver detalle de equipos
Descripción	El usuario quiere consultar equipos.
Datos de entrada	Se consulta un equipo.
Precondición	El usuario tiene que estar identificado, o se anonimo para consultar los equipos por defecto.
Datos de Salida	<p>El sistema carga un listado de los equipos por defecto, o un listado de los equipos por usuario. OK</p> <p>Se muestran los datos y jugadores de ese equipo. OK</p>

Instalación y configuración

Para la instalación y configuración del proyecto necesitaremos NodeJs ya que Angular CLI es una herramienta de este. Para ello tendremos que acceder a <https://nodejs.org> y descargar el instalador para nuestro sistema operativo.

Una vez tengamos NodeJs instalado ya podremos ejecutar los comandos npm, así que lanzaremos el comando **“npm install -g @angular/cli”**.

Durante el proceso de instalación se instalará el propio Angular CLI junto con todas sus dependencias. La instalación puede tardar varios minutos.

Una vez instalado dispondremos del comando “ng” a partir del cual se podrá lanzar cualquiera de las acciones que se puedan hacer mediante la interfaz de comandos de Angular.

Tras esto iremos al repositorio de git donde esta alojado el proyecto y en “Code” copiaremos el siguiente enlace:

- https://github.com/DavidPerezVelilla/TFG_DavidPerezVelilla.git

Podremos clonar el repositorio en la carpeta que queramos de nuestro equipo con el comando **“git clone https://github.com/DavidPerezVelilla/TFG_DavidPerezVelilla.git”**.

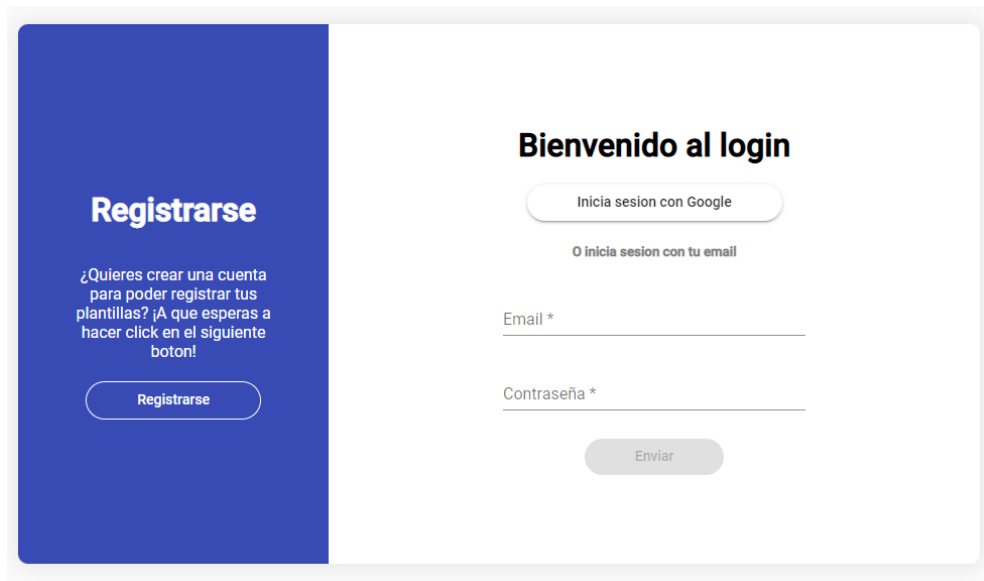
Una vez el proyecto esté descargado en nuestro equipo, accedemos a la carpeta contenedora y allí tendremos que ejecutar el comando **“npm install”** para instalar las dependencias necesarias para el correcto funcionamiento de la app.

Finalmente podremos ejecutar **“ng serve -o”** para lanzar el local nuestra aplicación, abriéndose una pestaña en nuestro navegador y mostrándonos la aplicación completamente funcional.

Manual de usuario

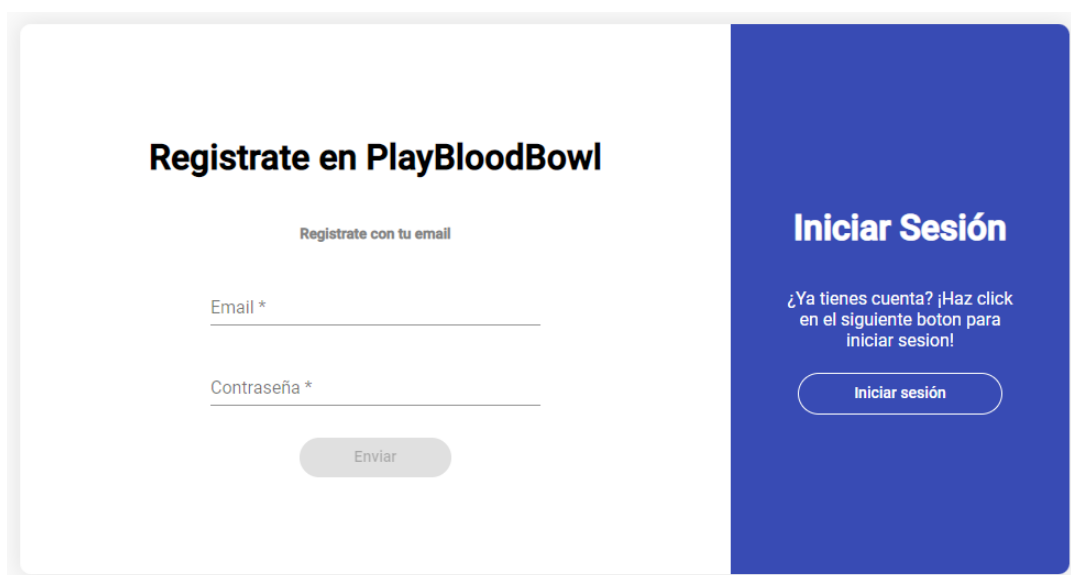
En este apartado trataré de explicar el funcionamiento y los pasos a seguir para el uso de PlayBloodBowl, pese a que la aplicación es fácil y sencilla de usar, ya que la idea es que sea muy intuitiva y accesible para todo el mundo.

Cuando accedemos a ella por primera vez se nos redirecciona al login. Aquí podemos iniciar sesión directamente con Google sin necesidad de registrarnos previamente, el sistema lo hará por nosotros.



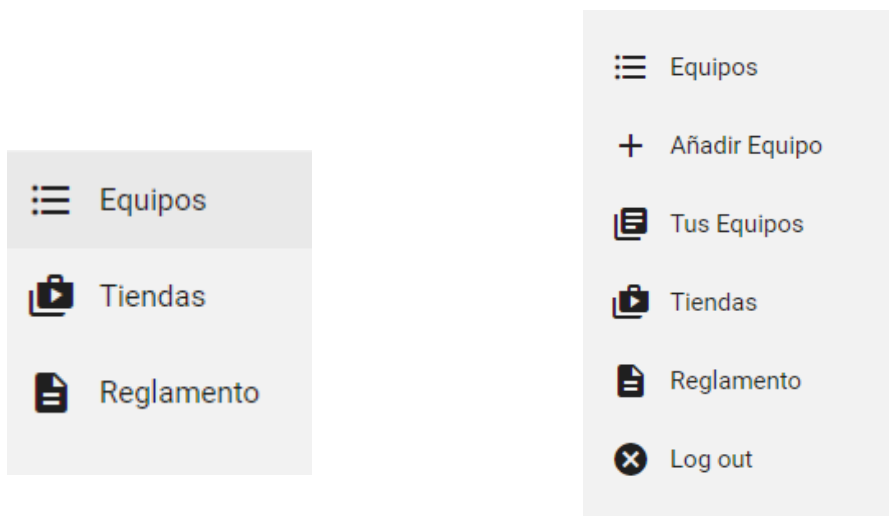
The image shows a login and registration interface. On the left, a blue vertical panel contains the text "Registrarse" and a message: "¿Quieres crear una cuenta para poder registrar tus plantillas? ¡A que esperas a hacer click en el siguiente boton!". Below this is a "Registrarse" button. On the right, a white panel contains the heading "Bienvenido al login". Below this are two buttons: "Inicia sesion con Google" and "O inicia sesion con tu email". Further down are input fields for "Email *" and "Contraseña *", followed by an "Enviar" button.

Si ya disponemos de cuenta también podemos iniciar sesión rellenando el formulario, habilitandose el botón de enviar cuando hayamos introducido los datos. Si lo que queremos es registrarnos podemos hacer click en registrarse, para que así nos muestre el formulario de registro, muy parecido al de login. Tendremos que añadir mínimo 6 caracteres a la contraseña.








The image shows a registration and login interface. On the left, a white panel contains the heading "Registrate en PlayBloodBowl". Below this is the text "Registrate con tu email". Further down are input fields for "Email *" and "Contraseña *", followed by an "Enviar" button. On the right, a blue vertical panel contains the heading "Iniciar Sesión". Below this is a message: "¿Ya tienes cuenta? ¡Haz click en el siguiente boton para iniciar sesion!". At the bottom of this panel is an "Iniciar sesión" button.


Tras iniciar sesión se nos redirigirá a la página de inicio y se habrán habilitado varias opciones que no están disponibles si no estamos identificados en el sistema.



Si queremos consultar los equipos por defecto tendremos que ir a **Equipos**, y se nos mostrará un listado con cada uno de ellos.

Escudo	Equipo	Descripción
	Dwarf Giants	Los Giants son un viejo equipo con una historia larga y distinguida. Fueron una de las sectas de Blood Bowl originales de Roze-El, y han estado presentes en todos los cambios y avances importantes que ha sufrido el juego. En tiempo recientes han consolidado su talento.
	Skavenblight Scramblers	Surgidos de las mas infectas alcantarillas, los Skavenblight son el principal equipo skaven del mundo. Ganador por 3 veces de la Blood Bowl, este equipo esta compuesto por una mezcla salvaje de la velocidad natural de los Skavens y el poderio fisico de sus Ratas Ogro.
	Nurgle's Rotters	Formado por infectados de forma atroz por una cepa de la Putrefacción de Nurgle que asolo al Imperio, los cuales buscaron el patronazgo del Dios de la Plaga (Nurgle)
	Elfheim Eagles	Cuando la Union Elfica se fundó, a sus primeros equipos les rechazaba el comportamiento opulento de sus antecesores y mas de un equipo se hundio cuando su plantilla decidio que necesitaban nuevos peinados de diseño. Asi surgieron los Eagles, liderando una nueva direccion para la Union Elfica.
	Los Naggarth Nightmares	Se les recuerda por ser uno de los primeros equipos Druchii en reclutar a muchos aspirantes entre el Culto a Khaine, en lugar de recurrir a las academias deportivas, y en hacer un buen uso de esos homicidas en roles ofensivos, una tactica que otros equipos de Elfos Oscuros adoptaron. Tienen gran rivalidad con los equipos de Ulthuan.

Pinchando en el escudo podremos consultar los jugadores que componen el equipo que queramos consultar, junto con sus atributos, coste...

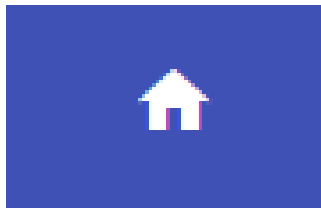


Dwarf Giants

Los Giants son un viejo equipo con una historia larga y distinguida. Fueron una de las sectas de Blood Bowl originales de Roze-El, y han estado presentes en todos los cambios y avances importantes que ha sufrido el juego. En tiempo recientes han consolidado su talento.

Posición	Movimiento	Fuerza	Agilidad	Pase	Armadura	Habilidades	Coste
Troll Slayer	5	3	2	0	8	Placar,Agallas,Furia,Cabeza Dura	95k
Dwarf Blocker Lineman	4	3	2	1	9	Placar,Placaje Defensivo,Cabeza Dura	70k
Dwarf Runner	6	3	3	2	8	Manos Seguras,Cabeza Dura	85k
Dwarf Blitzter	5	3	3	2	9	Placar,Cabeza Dura	80k
Deathroller	4	7	1	0	10	Abrirse Paso,Jugar Sucio (+2),Imparable,Solitario (5+),Golpe Mortifero (+1),Sin Manos,Arma Secreta,Mantenerse Firme	170k

Si todavía no estuviéramos logeados en el sistema y queremos volver al login, o estando logeados queremos volver a la página principal podemos hacer click en el icono de la casa que aparece en el navbar.



Para poder consultar las tiendas tendremos que hacer click en **Tiendas**, y se nos mostrarán unas tarjetas con las tiendas y su información. Podemos visitar su pagina web haciendo click en el botón que aparece debajo de la dirección.



Av. Cataluña 106, 50014,
Zaragoza

Odisea Games

Tienda de juegos de mesa y miniaturas donde encontraras todo lo necesario para iniciarte en cualquier juego de miniaturas. Ademas organizan ligas, torneos y concursos de pintura, de distintos juegos, creando una gran comunidad.

Si queremos consultar el reglamento, accederemos a él pinchando en Reglamento, donde veremos una breve información del juego y a parte un botón para descargar un pdf.

Reglamento

Todo aspirante a entrenador tiene que empezar por alguna parte. Aquí aprenderás conceptos básicos sobre el terreno de juego, los equipos, y las reglas.

En Blood Bowl, no solo importa lo que sucede durante el partido. Como entrenador, es tu deber conocer los puntos fuertes de tus jugadores, desde las habilidades a sus reglas especiales, y planear jugadas ganadoras. Para conocer esto y mucho mas pulsa en el siguiente botón.

Descargar

Para poder añadir equipos y gestionarlos, como ya he comentado, tendremos que estar identificados en el sistema. Para poder añadir un nuevo equipo nos tendremos que dirigir a **Añadir Equipos**. Aquí se nos mostrará un formulario que podremos rellenar, siendo obligatorios los tres primeros campos (Nombre del equipo, Descripción y Raza). Tras rellenar estos campos se habilitará el botón “Crear”, permitiéndonos añadir al equipo. Otra opción que tenemos es la de ir directamente a consultar nuestros equipos creados pulsando en el botón “Tus equipos”.

Una vez creado el equipo aparecerá un aviso en la parte inferior avisandonos, tendremos que darle a aceptar para cerrarlo.

Añadir

¿Quieres crear tus equipos y poder gestionarlos? ¡Rellena el siguiente formulario para ello! Si quieres consultar tus equipos haz click en el boton.

Tus equipos

Nombre del equipo *

Equipo de muestra

Descripcion *

Esto es una explicacion

Raza *

Icono (url)

Imagen (url)

Crear

Añadir

¿Quieres crear tus equipos y poder gestionarlos? ¡Rellena el siguiente formulario para ello! Si quieres consultar tus equipos haz click en el boton.

Tus equipos

Nombre del equipo *

Equipo de muestra

Descripcion *

Esto es una explicacion

Elfos Oscuros

Caos

Enanos

Humanos

No Muertos

Elfos

Icono (url)

Imagen (url)

Crear

Equipo creado con éxito!!

Aceptar

Ahora, puesto que ya tenemos un equipo creado podemos administrarlo, para ello accederemos a través del sidenav pulsando en **Tus Equipos**, o como ya he mencionado anteriormente, a través del botón que aparece al lado del formulario de creación. Aquí se mostrará una lista de nuestros equipos (si no tenemos ninguno aparece vacía).

Escudo	Equipo	Descripción	Opciones
	Equipo de muestra	Esto es una explicación	<button>Consultar</button> <button>Borrar</button>

Podremos borrarlo o consultarlo. Si hacemos click en Consultar el sistema nos mostrara una tarjeta de equipo como la de los equipos por defecto.

Equipo de muestra
 Esto es una explicación

Posición Movimiento Fuerza Agilidad Pase Armadura Habilidades Coste

Jugadores *

Desde ella podemos añadir jugadores a nuestro equipo, que se nos mostrarán en un desplegable dependiendo de la raza que hayamos seleccionado anteriormente. Tras añadir un jugador nos saldrá un aviso y este se mostrará en la tabla junto sus atributos.

Posición Movimiento Fuerza Agilidad Pase Armadura Habilidades Coste

Rotter Lineman
 Rotspawn
 Pestigor
 Bloater

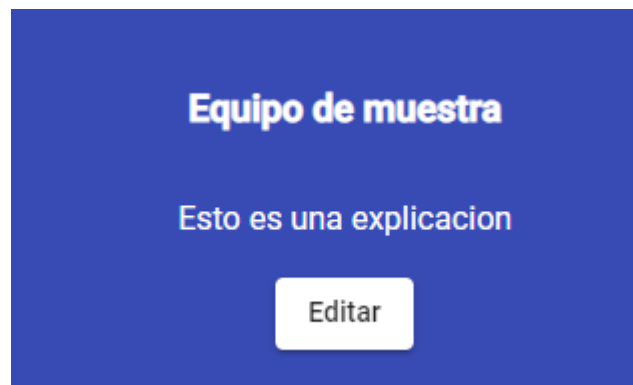
Jugadores *
Bloater

Posición	Movimiento	Fuerza	Agilidad	Pase	Armadura	Habilidades	Coste
Bloater	4	4	2	1	9	Presencia Perturbadora, Apariencia Asquerosa, Infectado, Regeneración	115k

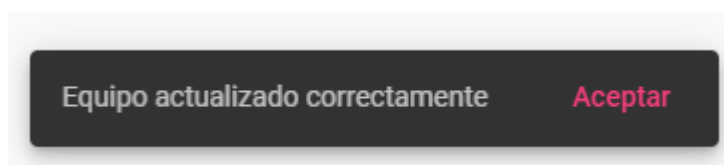
Jugadores *

Jugador añadido correctamente

Si lo que queremos es editar algún campo del equipo lo podremos hacer pulsando en el botón editar.



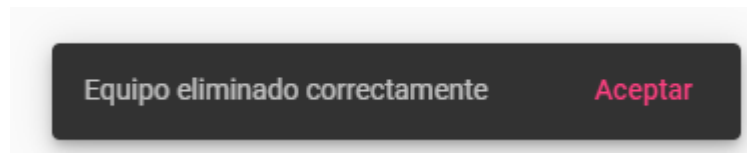
Nuevamente esto nos llevará a un nuevo formulario en el que podremos modificar los campos anteriores, aunque esta vez no podremos cambiar la raza y ningún campo es obligatorio. Tras darle al botón actualizar nos saldrá un nuevo aviso indicándonos que hemos tenido éxito. Si volvemos a consultar nuestros equipos veremos como descripción se ha actualizado correctamente.

A form for editing a team. On the left is a blue sidebar with the title "Editar" in white. Below the title is the text "¿Quieres editar tu equipo? ¡Rellena el siguiente formulario para ello! Si quieres consultar tus equipos haz click en el boton." and a white button with the text "Tus equipos". On the right is a white form area with three input fields: "Nombre del equipo" with the value "Equipo de muestra", "Descripcion" with the value "Esto es una modificacion", and "Icono (url)". Below these fields is a white button with the text "Actualizar".

Escudo	Equipo	Descripcion	Opciones	
	Equipo de muestra	Esto es una modificacion	Consultar	Borrar

Para un equipo, desde esta página simplemente tendremos que hacer click en Borrar. Tras ello se nos mostrará un nuevo mensaje informándonos y se nos redirigirá a la página de inicio.

Escudo	Equipo	Descripción	Opciones
	Equipo de muestra	Esto es una modificación	Consultar Borrar



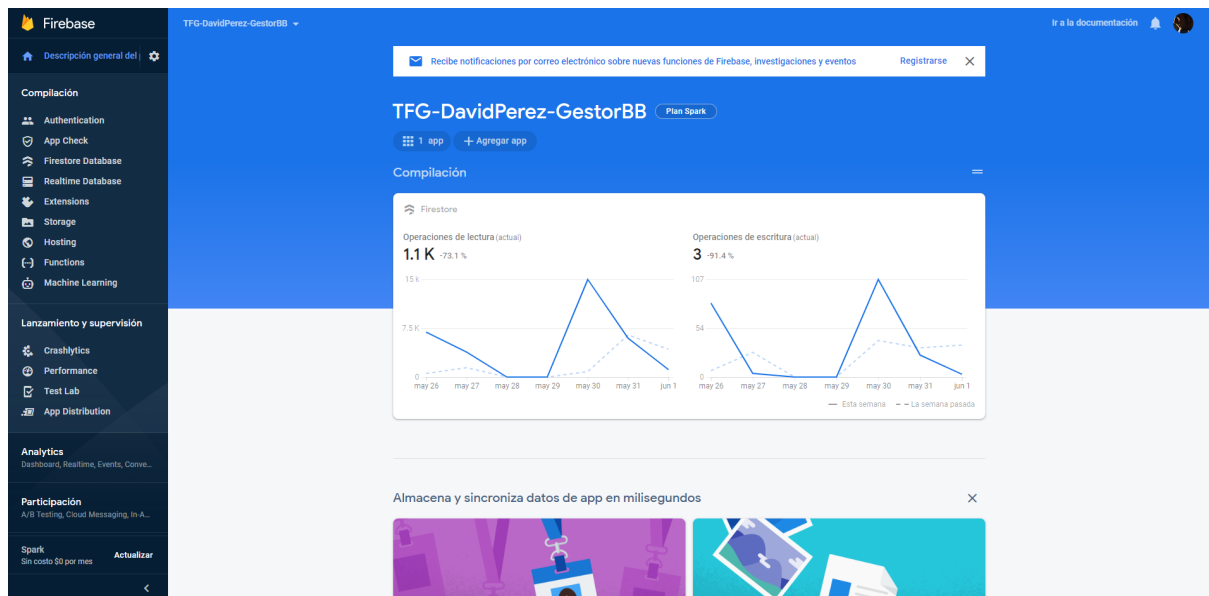
Escudo	Equipo	Descripción	Opciones
--------	--------	-------------	----------

Finalmente, para cerrar sesión pinchamos sobre **Logout**, que cerrará nuestra sesión y nos llevará al login, haciendo desaparecer también del sidenav algunas de las opciones.

Para gestionar el panel de Administrador, en este caso la consola de Firebase, hay que acceder con mi cuenta personal de Google, por lo que solo está permitido mi acceso. Se nos mostrarán los proyectos disponibles.



Después de seleccionar el proyecto TFG, entraremos al panel de Administrador, donde podremos ver las operaciones de lectura y escritura, diversos tutoriales para el uso de Firebase y distintas opciones de gestión, como pueden ser Authentication, donde veremos los usuarios que se han dado de alta en PlayBloodBowl, o Firestore Database, la base de datos de nuestra aplicación.



Buscar por dirección de correo electrónico, número de teléfono o UID de usuario

Agregar usuario

Identificador	Proveedores	Fecha de creación	Fecha de acceso	UID de usuario
perico@outlook.com		26 may 2022	26 may 2022	2ZUd7JJ0n2QAYxpxGmn6Oxn75k...
tenno926@gmail.com		24 may 2022	1 jun 2022	VPSijQC8EbZ1SU2JdvZhrMlxm1
dvdpv926@gmail.com		20 may 2022	2 jun 2022	FTztAfVJDgMnSsvLSvPf1jkVFqG2

Filas por página: 50 1 - 3 of 3

Cloud Firestore

Datos Reglas Índices Uso

Protege tus recursos de Cloud Firestore contra los abusos, como fraudes de facturación o suplantación de identidad. Configurar la Verificación de aplicaciones

Crea prototipos y haz pruebas de extremo a extremo con Local Emulator Suite, que ahora es compatible con Firebase Authentication. Comenzar

player > 0HKoebmLcbKr...

tf-g-davidperez-gestorbb player 0HKoebmLcbKr5ogqtek

+ Iniciar colección + Agregar documento + Iniciar colección + Agregar campo

player > 0HKoebmLcbKr5ogqtek >

race
sub_race
team

2092Pqt0rYE0VY7Q8SpI
2jA8QDK2fnT2Zht3Ac3k
2pyLRpSUZJ04rDxk8wHB
2tuYVqBqpsnSBjwFyCF6
3nMxNjPSXv17a5inLnLv
4Z1jaFMeAv0uUrPUCNS
6KIoaASWqLmbdAJeKN0w
7At7oam3tTzPaqKK2TUc
8ajdZ1bLScVBonyXt3Lr
9I2DiC9y9zhChpS8zvDv
9MmLIFwUXSPuS8RIAF
AGtpQXXPU0DqEp2BRAtm

agl: 3
arm: 8
cost: "90k"
hab
"Mantenerse Firme"
"Forcejea"
mov: 6
player_name: "Bodyguard"
race: "oDX6kCod5TLJDwXn4Tly"
str: 6
team: "IRdrED8aShTXRK8IuIBC"
thr: 1

Sobre la base de datos podemos ver las colecciones que la componen y los documentos pertenecientes a esas colecciones, en este caso la colección “player” que contiene los jugadores de los equipos. Cada jugador consta de una serie de atributos.

Podemos hacer cualquier consulta gracias a un buscador que ejecutará estas consultas de una manera muy simple y fácil de usar, con una interfaz gráfica, sin necesidad de tener que escribirla nosotros.

Filtrar por campo ^

Ingresar campo

Campos sugeridos

agl

arm

cost

hab

mov

player_name

race

str

team

thr

Agregar condición v

Ordenar resultados v

.collection("player")

Borrar

Cancelar

Aplicar

Filtrar por campo ^

player_name x

Agregar condición "==" "Bloater" v

Ordenar resultados v

.collection("player")
.where("player_name", "==", "Bloater")

También podemos agregar, borrar, o borrar todos los datos de una colección o documento de manera sencilla.

tfg-davidperez-gestorbb	player
+ Iniciar colección	+ Agregar documento

Tfg6c2JkkJEXn6soxjjM	<div> <div>Borrar documento</div> <div>Borrar los campos del documento</div> </div>
+ Iniciar colección	
+ Agregar campo	

Agregar un documento

Ruta superior
/player

ID de documento ?

ID automático

! Obligatorio

Campo	Tipo	Valor
<input type="text"/>	= string	<input type="text"/>
<div><div>+</div></div>		

Cancelar

Guardar

Inicia una colección

1

Asignar un ID a la colección

2

Agregar el primer documento

Ruta superior
/

ID de la colección ?

Cancelar

Siguiente

Como vemos es una herramienta muy potente que nos permite a través de una interfaz gráfica generar todas estas consultas, facilitando al usuario y haciendo esta gestión mucho más rápida y eficiente.

En cuanto a la gestión de usuarios, podemos gestionar los usuarios, agregar nuevos, dar de baja... A parte que veremos toda la información relevante sobre ellos.

<div> <div> <div>🔍</div> <div> Buscar por dirección de correo electrónico, número de teléfono o UID de usuario </div> </div> <div> <div>Agregar usuario</div> <div>↻</div> <div>⋮</div> </div> </div>					
Identificador	Proveedores	Fecha de creación	↓	Fecha de acceso	UID de usuario
perico@outlook.com	✉	26 may 2022		26 may 2022	2ZUd7JJ0n2QAYxpxGmn6Oxn75k...
tenno926@gmail.com	✉	24 may 2022		1 jun 2022	VPSjjQC8EbFzh1SU2JdvZhrMlxm1
dvdpv926@gmail.com	🌐	20 may 2022		2 jun 2022	FTztAfVJDgMnSsvLSvPf1jkVFqG2



Agrega un usuario con correo electrónico y contraseña

Correo electrónico

Contraseña

Cancelar

Agregar usuario

Correo electrónico	Estado	Fecha de creación	Fecha de actualización	Acciones
perico@outlook.com		26 may 2022	26 may 2022	Restablecer contraseña
tenno926@gmail.com		24 may 2022	1 jun 2022	Inhabilitar cuenta
duday206@gmail.com		22 may 2022	2 jun 2022	Borrar cuenta

Además de poder acceder a través de registro o Google, Firebase también nos permite el registro con otros proveedores, pudiendo hacerlo con una cuenta de Yahoo, GitHub, Facebook, entre muchos otros.

Proveedores de acceso


Agregar proveedor nuevo


Proveedor


Estado

Selecciona un proveedor de acceso (paso 1 de 2)


Proveedores nativos


 Correo electrónico/contraseña ✓


 Teléfono


 Anónimo


Proveedores adicionales


 Google ✓


 Facebook


 Play Juegos


 Game Center


 Apple


 GitHub


 Microsoft


 Twitter

 Yahoo

 Correo electrónico/contraseña

 Habilitado

 Google

 Habilitado

También nos permite agregar dominios.

Domínios autorizados ?

[Agregar un dominio](#)

Dominio autorizado	Tipo
localhost	Default
tfg-davidperez-gestorbb.firebaseio.com	Default
tfg-davidperez-gestorbb.web.app	Default

Otra de sus características es que ofrece plantillas para la gestión de recuperación de contraseñas, verificación de correo electrónico...

Plantillas

Correo electrónico ?

- Verificación de dirección de correo...**
- Restablecer contraseña
- Cambio de dirección de correo ele...
- Configuración del SMTP

SMS

- Verificación por SMS

Verificación de dirección de correo electrónico

Cuando un usuario se registra con una dirección de correo electrónico y una contraseña, puedes enviarle un mensaje de confirmación para verificar la dirección de correo electrónico que registró. [Más información](#)

Nombre del remitente

De

no proporcionado

noreply@tfg-davidperez-gestorbb.firebaseio.com

Responder a

noreply

Asunto

Verify your email for %APP_NAME%

Mensaje

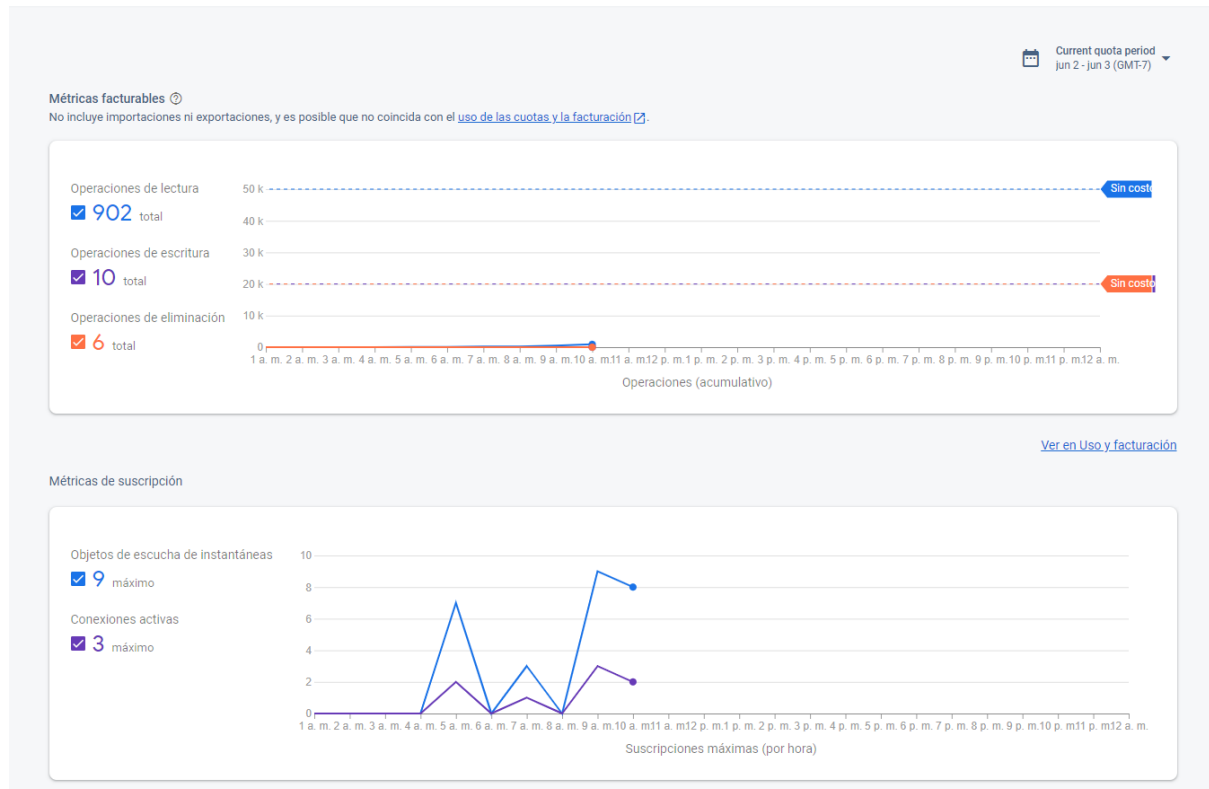
Hello %DISPLAY_NAME%,
Follow this link to verify your email address.
https://tfg-davidperez-gestorbb.firebaseio.com/_/auth/action?mode=action&oobCode=code
If you didn't ask to verify this address, you can ignore this email.
Thanks,
Your %APP_NAME% team

Idioma de la plantilla

inglés

Así que para un proyecto como este, es una herramienta estupenda para la gestión de usuarios dada las facilidades que aporta.

Para terminar, FireBase también nos ofrece información sobre nuestra aplicación, pudiendo consultar estos datos para saber si nuestra web es visitada o no, las operaciones que se están llevando a cabo... Lo que nos ayuda a poder gestionar mucho mejor nuestro proyecto.



En definitiva, el panel de administrador que nos brinda FireBase es bastante completo, fácil e intuitivo de usar, aportando un gran cantidad de herramientas muy útiles para la gestión de cualquier proyecto, aunque para proyectos más grandes deberíamos usar la versión de pago ya que la gratuita puede que llegue a quedarse corta.

4. Documento de cierre

Resultados obtenidos y conclusiones

La realización de este Trabajo de Fin de Ciclo contribuye a demostrar las habilidades adquiridas a lo largo de estos dos años de formación en el Ciclo de Desarrollo de Aplicaciones Web.

Su desarrollo además ha sido una estupenda formación en Angular, framework que usare en la empresa tras mi periodo de prácticas. Me ha servido para comprender mejor cómo funciona, cómo se relacionan los distintos componentes de Angular entre sí, e irme familiarizando con él, pese a que se trata de una aplicación sencilla y todavía me queda mucho que aprender de Angular. También me ha servido para conocer las bases de datos no relacionales, pues hasta ahora no conocía ninguna, por lo que durante el proceso he tenido que irme formando y aprendiendo.

Por último, el trabajar con una metodología ágil, y al contar con el apoyo de mi tutor de prácticas para concertar reuniones de como iba el proyecto, de los sprints a realizar y las tareas a llevar a cabo en cada uno de ellos (siempre organizado todo por mí) me ha ayudado a comprender cómo es el día a día de un programador dentro de la empresa, trabajando por tareas y teniendo que desarrollarlas para estar disponibles en x tiempo, el gestionar y organizar un proyecto con todo lo que ello conlleva.












El resultado final es lo que esperaba, siendo mejorable conforme tenga más conocimientos, pues en este mundo tienes que estar en continua formación, aprendiendo nuevos conceptos cada día. Pero estoy satisfecho con funcionamiento, siendo la base para un proyecto mucho mayor que tengo pensado llevar a cabo, como ya he comentado al principio de este trabajo. Todavía quedan muchas horas de desarrollo y de aprendizaje, nuevos componentes que añadir, otros que suprimir, mejorar la interfaz y el diseño, pero a día de hoy, es una aplicación completamente funcional, que como ya digo, es el primer paso para un proyecto mucho más ambicioso.

Diario de bitácora

Para la realización del diario de bitácora se han tenido en cuenta los commits realizados en GitHub en el repositorio del proyecto, siempre en la rama main.

Repositorio: https://github.com/DavidPerezVelilla/TFG_DavidPerezVelilla

Commits on Jun 2, 2022	<div>tests David Perez Velilla committed 2 hours ago</div> <div>fa47aed</div>
Commits on Jun 1, 2022	<div>correcciones David Perez Velilla committed yesterday</div> <div>9a6845b</div> <div>edit final David Perez Velilla committed yesterday</div> <div>1055e70</div>
Commits on May 31, 2022	<div>final David Perez Velilla committed 2 days ago</div> <div>fd550cd</div> <div>store David Perez Velilla committed 2 days ago</div> <div>79ae1bb</div>
Commits on May 30, 2022	<div>edit detail David Perez Velilla committed 3 days ago</div> <div>f062cf8</div>
Commits on May 27, 2022	<div>filtros David Perez Velilla committed 6 days ago</div> <div>61e265e</div>
Commits on May 26, 2022	<div>arreglos maquetacion David Perez Velilla committed 7 days ago</div> <div>5b6be1d</div> <div>usuario-equipos David Perez Velilla committed 7 days ago</div> <div>d715be1</div>
Commits on May 25, 2022	<div>rules David Perez Velilla committed 8 days ago</div> <div>40bd6da</div>
Commits on May 24, 2022	<div>detail style David Perez Velilla committed 9 days ago</div> <div>882d61e</div> <div>detail teams David Perez Velilla committed 9 days ago</div> <div>e8f67de</div>
Commits on May 20, 2022	<div>login David Perez Velilla committed 13 days ago</div> <div>7a9df22</div>
Commits on May 17, 2022	<div>new models David Perez Velilla committed 16 days ago</div> <div>e6f55eb</div>

service	David Perez Velilla committed 22 days ago	 cbc6422 <>
interfaces	David Perez Velilla committed 22 days ago	 a3a5f2f <>
Firebase	David Perez Velilla committed 22 days ago	 d6aFa3b <>
plantilla	David Perez Velilla committed 22 days ago	 dFdd2ec <>
material	David Perez Velilla committed 22 days ago	 40456d0 <>
routing module	David Perez Velilla committed 22 days ago	 191d65d <>
Merge branch 'main' of https://github.com/DavidPerezVelilla/TFG_David...	David Perez Velilla committed 22 days ago	 100b44d <>
creacion components	David Perez Velilla committed 22 days ago	 a762f27 <>
Update README.md	 DavidPerezVelilla committed 22 days ago	 6748f6a <>
initial commit	David Perez Velilla committed 22 days ago	 89ac556 <>

5. Bibliografía

- <https://www.adictosaltrabajo.com/2017/09/05/angular-material-firebase/>
- <https://www.positronx.io/create-angular-firebase-crud-app-with-angular-material/>
- <https://www.becomebetterprogrammer.com/angular-signup-page-with-firebase-authentication/>
- <https://angular.io/>
- <https://material.angular.io/>
- <https://firebase.google.com/docs/firestore/quickstart>
- <https://github.com/>
- <https://nodejs.org/en/>
- <https://app.pluralsight.com/>